

Azure Command-Line Interface (CLI) documentation

The Azure command-line interface (Azure CLI) is a set of commands used to create and manage Azure resources. The Azure CLI is available across Azure services and is designed to get you working quickly with Azure, with an emphasis on automation.

Start here

OVERVIEW

[What is Azure CLI?](#)

[Get started](#)

[Onboarding cheat sheet](#)

[Terminology and support levels](#)

Install

DOWNLOAD

[Install - Windows](#)

[Install - macOS](#)

[Install - Linux](#)

[Run in Azure Cloud Shell ↗](#)

[Run in a Docker container](#)

Sign in

HOW-TO GUIDE

[Authentication methods](#)

[Work with service principals](#)

WHAT'S NEW

[Sign in with Web Account Manager \(WAM\)](#)

[Azure AD Graph to Microsoft Graph migration](#)

Next steps



TRAINING

[Learn to use the Azure CLI tutorial](#)

[How-to use Azure CLI in a Bash environment](#)

[Considerations for using Azure CLI in a PowerShell environment](#)

[Tips to use Azure CLI successfully](#)

[Script to create Azure resources at scale](#)

Find documentation



REFERENCE

[Reference index A to Z](#)

[Article \(docs\) index A to Z](#)

[Sample index A to Z](#)

What's new



WHAT'S NEW

[What's new highlights](#)

[Release notes](#)

How-to guides



CONCEPT

[Protect sensitive information](#)

[Configure settings](#)

[Change output formats](#)

[Query command output](#)

[Work in interactive mode](#)

[Work in multiple clouds](#)

[Use the Azure REST API](#)

[Manage Azure subscriptions](#)

[Manage Azure resource groups](#)

Manage cloud resources with the Azure CLI

TRAINING

[Create Azure resources](#)

[Control Azure services](#)

[Manage virtual machines](#)

[Connect an application to Azure Storage](#)

[Run parallel tasks in Azure Batch](#)

TUTORIAL

[Learn to use the Azure CLI](#)

[Create a virtual network and machine](#)

[Create an Azure service principal](#)

Give feedback and get support

CONCEPT

[Request a new feature ↗](#)

[Report a product bug ↗](#)

[Get help from the community ↗](#)

[Request support using the Azure CLI](#)

What is the Azure CLI?

Article • 03/07/2024

The Azure Command-Line Interface (CLI) is a cross-platform command-line tool to connect to Azure and execute administrative commands on Azure resources. It allows the execution of commands through a terminal using interactive command-line prompts or a script.

For interactive use, you first launch a shell such as cmd.exe on Windows, or Bash on Linux or macOS, and then issue a command at the shell prompt. To automate repetitive tasks, you assemble the CLI commands into a shell script using the script syntax of your chosen shell, and then you execute the script.

You can [install the Azure CLI](#) locally on Linux, macOS, or Windows computers. It can also be used from a browser through the [Azure Cloud Shell](#) or run from inside a Docker container.

Current Version

The current version of the Azure CLI is **2.58.0**. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run `az version`.

Authentication

Azure CLI supports several authentication methods. For detailed information about authenticating to Azure from the Azure CLI, see [Sign in with Azure CLI](#).

Command reference design

The Azure CLI syntax follows a simple `reference name - command - parameter - parameter value` pattern. For example, switching between subscriptions is often a common task. Here's the syntax.

Azure CLI

```
az account set --subscription "my subscription name"
```

Another common use of the Azure CLI is managing role assignments.

Azure CLI

```
az role assignment create --assignee servicePrincipalName --role Reader --  
scope /subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName  
az role assignment delete --assignee userSign-inName --role Contributor
```

For more information on managing subscriptions, see [Manage subscriptions with Azure CLI](#). See [Create an Azure service principal with the Azure CLI](#) for an in-depth tutorial on managing service principals and role assignments.

PowerShell syntax comparison

[Choose the right command-line tool](#) explains the difference between `tools` and `environments` with an emphasis on the Azure CLI and Azure PowerShell. It also gives many [side-by-side command comparisons](#). Here are two examples:

[+] [Expand table](#)

Command	Azure CLI	Azure PowerShell
Create Resource Group	az group create --name <ResourceGroupName> --location eastus	New-AzResourceGroup -Name <ResourceGroupName> -Location eastus
Create Azure Storage Account	az storage account create --name <StorageAccountName> --resource-group <ResourceGroupName> --location eastus --sku Standard_LRS --kind StorageV2	New-AzStorageAccount -Name <StorageAccountName> -ResourceGroupName <ResourceGroupName> -Location eastus -SkuName Standard_LRS -Kind StorageV2

For Azure CLI syntax comparisons between Bash and PowerShell environments, see [Learn Azure CLI syntax differences](#).

Output formats

The Azure CLI uses JSON as its default output format, but offers other formats as described in [Output formats for Azure CLI commands](#). Use the `--output` parameter to format Azure CLI command results. Here's an example:

Azure CLI

```
az account list --output table
```

Set your default output by setting the `output` configuration property as described in [Azure CLI configuration](#).

```
Azure CLI
```

```
az config set core.output=jsonc
```

Data collection

Azure CLI collects telemetry data by default. Microsoft aggregates collected data to identify patterns of usage to identify common issues and to improve the experience of Azure CLI. Microsoft Azure CLI doesn't collect any private or personal data. For example, the usage data helps identify issues such as commands with low success and helps prioritize our work.

While we appreciate the insights this data provides, we also understand that not everyone wants to send usage data. You can disable data collection with the `az config set core.collect_telemetry=false` command. You can also read our [privacy statement](#) ↗ to learn more.

See Also

- [Onboarding Cheat Sheet](#)
- [Get started with the Azure CLI](#)
- [Full command reference list for the Azure CLI](#)

Get started with Azure CLI

Article • 03/07/2024

Welcome to the Azure Command-Line Interface (CLI)! This article introduces the CLI and helps you complete common tasks.

Install or run in Azure Cloud Shell

The easiest way to learn how to use the Azure CLI is by running a Bash environment in [Azure Cloud Shell](#) through your browser. To learn about Cloud Shell, see [Quickstart for Bash in Azure Cloud Shell](#).

When you're ready to install the Azure CLI, see the [installation instructions](#) for Windows, Linux, macOS, and Docker container. If you're using the Azure classic deployment model, [install the Azure classic CLI](#).

Check your version by running `az --version`. Azure Cloud Shell always has the latest version of the Azure CLI preinstalled.

```
Azure CLI
az version
```

Sign into the Azure CLI

Before using any Azure CLI commands with a local install, you need to sign in with [az login](#).

1. Run the `az login` command.

```
Azure CLI
az login
```

If the Azure CLI can open your default browser, it initiates [authorization code flow](#) and opens the default browser to load an Azure sign-in page.

Otherwise, it initiates the [device code flow](#) and instructs you to open a browser page at <https://aka.ms/devicelogin>. Then, enter the code displayed in your terminal.

If no web browser is available or the web browser fails to open, you may force device code flow with `az login --use-device-code`.

2. Sign in with your account credentials in the browser.

After logging in, you receive a list of subscriptions associated with your Azure account. The subscription information with `isDefault: true` is the currently activated subscription after logging in. To select another subscription, use the [az account set](#) command with the subscription ID of the desired account. For more information about subscription selection, see [Manage Azure subscriptions](#).

There are ways to sign in non-interactively, which are covered in detail in [Sign in with Azure CLI](#).

Find commands

Azure CLI commands are organized as command groups. Each group represents an area of an Azure service. There are two options to find command groups:

- Use the `az find` command. For example, to search for command names containing `vm`, use the following command:

```
Azure CLI
```

```
az find vm
```

- Use the `--help` argument to get a complete list of subgroups within a reference group. This example returns all the subgroups for virtual machines:

```
Azure CLI
```

```
az vm --help
```

Here is example output with lines omitted for brevity:

```
Output
```

```
Subgroups:
```

application	: Manage applications for VM.
availability-set	: Group resources into availability sets.
boot-diagnostics	: Troubleshoot the startup of an Azure Virtual Machine.
...	

Here's another example that finds the Azure CLI commands for grouping virtual machines into availability sets, a *subgroup* of `az vm`:

```
Azure CLI
```

```
az vm availability-set --help
```

Also use `--help` to get parameter lists and command examples for a reference *command*.

```
Azure CLI
```

```
az vm create --help
```

Here is example output with lines omitted for brevity:

```
Output
```

Arguments

```
--name [Required] : Name of the virtual machine.
```

```
...
```

Authentication Arguments

```
--admin-password : Password for the VM if authentication type is  
'Password'.
```

```
--admin-username : Username for the VM...
```

```
...
```

Managed Service Identity Arguments

```
...
```

Examples

```
Create a VM from a custom managed image.
```

```
az vm create -g MyResourceGroup -n MyVm --image MyImage
```

```
...
```

- Use the [A to Z reference index](#) that lists all command groups alphabetically.

Find published samples and articles

If you prefer to learn Azure CLI syntax by seeing tested published examples, there are two indexes of article content:

- [Samples index A to Z](#). The samples index provides a list of Azure CLI samples sorted by [subject area](#), [reference group](#), and [GitHub repository](#).
- [Article index A to Z](#). Use your keyboard `find` shortcut keys, like `ctrl + F`, to drop quickly to the reference command group in which you're interested. For example,

the article index looks something like this for `az vm`:

 Expand table

Reference subgroup	Azure CLI article showing reference use
<code>az vm</code>	Output formats for Azure CLI commands
	How to use variables in Azure CLI commands
	Get VM information with queries
	{More articles listed here.}
<code>az vm aem</code>	New Version of Azure VM extension for SAP solutions
	Standard Version of Azure VM extension for SAP solutions
<code>az vm application</code>	{...}

Use tab completion

The CLI has full tab completion for commands in a Bash environment. To enable tab completion in a PowerShell environment, see [Enable tab completion in PowerShell](#).

Be aware of globally available arguments

There are some arguments that are available for most commands.

- `--help` prints CLI reference information about commands and their arguments and lists available subgroups and commands.
- `--output` changes the output format. The available output formats are `json`, `jsonc` (colorized JSON), `tsv` (Tab-Separated Values), `table` (human-readable ASCII tables), and `yaml`. By default the CLI outputs `json`. To learn more about the available output formats, see [Output formats for Azure CLI](#).
- `--query` uses the [JMESPath query language](#) to filter the output returned from Azure services. To learn more about queries, see [Query command results with Azure CLI](#) and the [JMESPath tutorial](#).
- `--verbose` prints information about resources created in Azure during an operation, and other useful information.

- `--debug` prints even more information about CLI operations, used for debugging purposes. If you find a bug, provide output generated with the `--debug` flag on when submitting a bug report.
- `--subscription` is the name or ID of a subscription. For more information about setting your default subscription, see [Manage Azure subscriptions with the Azure CLI](#).
- `--only-show-errors` suppresses warnings in command output and only shows errors.

Use interactive mode

The CLI offers an interactive mode that automatically displays help information and makes it easier to select subcommands. You enter interactive mode with the [az interactive](#) command.

```
Azure CLI
az interactive
```

For more information on interactive mode, see [Run the Azure CLI in Interactive Mode](#).

There's also a [Visual Studio Code plugin](#) that offers an interactive experience, including autocomplete and mouse-over documentation.

Learn Azure CLI basics with quickstarts and tutorials

To learn how to use the Azure CLI, try our [onboarding tutorial](#) that teaches you the following skills while working with Azure Storage:

- Find and change your default subscription.
- Create an Azure resource containing a random ID in the name.
- Set environment variables.
- Learn syntax differences between Bash, PowerShell, and Cmd.
- Debug Azure CLI reference commands.
- Get properties from a local JSON file.. This example shows how to get configuration values for development, stage, and production.
- Delete multiple resources using a script.

Note

In scripts and on the Microsoft documentation site, Azure CLI examples are written for the Bash shell. Most one-line examples will run on any platform. Longer examples which include line continuations (\), variable assignment, or quotes need to be modified to work in other shells, including PowerShell. For more information on syntax differences between environments, see [Learn syntax differences between Bash, PowerShell, and Cmd.](#)

Give feedback

We welcome your feedback for the CLI to help us make improvements and resolve bugs. You can [file an issue on GitHub](#) or use the built-in features of the CLI to leave general feedback with the `az feedback` command.

Azure CLI

```
az feedback
```

See also

- [Onboarding cheat sheet](#)
- [Learn to use the Azure CLI tutorial](#)
- [Use the Azure CLI in a Bash environment](#)
- [Considerations for using the Azure CLI in a PowerShell environment](#)

Azure CLI onboarding cheat sheet

Article • 06/27/2024

If you're looking to onboard quickly with the Azure CLI, you have come to the right place! Use this page to learn tips and techniques when getting started with the Azure CLI.

Content indexes

Jump straight to reference command examples, sample scripts, and tutorials using these links:

- Find an index of quickstarts, how-to guides and tutorials in [Azure CLI doc index A-Z](#).
- Choose a tested script and make it your own from the [Azure CLI sample index A-Z](#).
- For a published format of in-line help, go to the [Reference index A-Z](#).
- Go to [Microsoft Learn Modules for CLI tools](#) for free on-line training with a sandbox environment.

Getting started

[] Expand table

Question	Answer	Learn more
What advantage is there to using Azure CLI?	You can manage multiple Azure resources concurrently from a flexible command-line. For example, create 50 Azure storage accounts, or update multiple user permissions through a script.	Choose the right command-line tool
Where can I run the Azure CLI?	The Azure CLI runs in Windows (Cmd or PowerShell), Linux, macOS, Docker containers, and Azure Cloud Shell. Using Azure Cloud Shell is the quickest way to get started.	Run Azure Cloud Shell
Do I have to install the Azure CLI?	Azure Cloud Shell and some Linux distributions have the Azure CLI preinstalled. For all other environments, you must install the Azure CLI.	Install the Azure CLI
How do I run the Azure CLI in a Docker container?	<pre>docker run -it mcr.microsoft.com/azure-cli</pre>	How to run the Azure CLI in a Docker container

Question	Answer	Learn more
What URLs do I need for proxy bypass?	This depends on the Cloud you are using. For a complete list see the list of endpoints needed for proxy bypass.	Azure CLI endpoints for proxy bypass
How do I sign in?	Use <code>az login</code> inside your console window. Use your own Azure username and password, a service principal, managed identity or WAM.	Sign in with Azure CLI

Using reference commands

[\[+\] Expand table](#)

Question	Answer	Learn more
What is the Azure CLI syntax pattern?	"command group + <i>command subgroup</i> + command + --parameters" Example: "az storage account create --name myStorageAccount"	Azure CLI syntax components
What is the difference between a command group, cmdlet, command, alias, module, and reference?	Good question! The Azure CLI only has command groups, reference commands, and user-defined aliases. You may be thinking of Azure PowerShell if you are looking for cmdlets and modules.	Azure CLI syntax components
What is an extension and why does it have to be installed separately?	Extensions are command group add-ins that are not shipped as part of the core Azure CLI. You are prompted to install an extension the first time you use it. Get a list of available extensions by running <code>az extension list-available -output table</code> .	Use and manage extensions with the Azure CLI
How do I get a list of all commands, both core and extension?	For a list of command groups, in your console type <code>az</code> and <code>Enter</code> . For a list of subgroups and commands, use the <code>--help</code> parameter. Example: <code>az account --help</code> or <code>az account subscription --help</code> outputs a command list for just the group or subgroup.	Reference index A-Z
Does the Azure CLI have tab completion?	Yes! When typing a command in PowerShell, press your <code>Tab</code> key. This feature is only available in PowerShell.	Enable Tab Completion on PowerShell
How do I get in-line help?	Every command has a <code>--help</code> parameter that results in descriptions and tips to display in your console. Example: <code>az account --help</code> or <code>az vm create --help</code>	You can also Request support .

Question	Answer	Learn more
How can I learn to use multiple reference commands to complete a single job to be done?	We're glad you asked! The Azure CLI provides several interactive commands that provide syntax and next steps. In the next section, learn about <code>az interactive</code> , <code>az scenario</code> , <code>az next</code> , and <code>az find</code> .	Learning Azure CLI

For a side-by-side comparison of Azure CLI and Azure PowerShell syntax, see [Azure CLI vs Azure PowerShell: Side-by-side Command Comparison](#).

See script comparisons between Bash, PowerShell, and Cmd in [Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#).

Learning Azure CLI

The [Learn to use the Azure CLI tutorial](#) walks you through many common Azure CLI tasks with examples in Bash, PowerShell and Cmd. Here are some of the skills covered while working with Azure Storage:

- Decide what environment I should use
- Find and change my default subscription
- Create an Azure resource containing a random ID in the name
- Set environment variables
- Learn syntax differences between Bash, PowerShell and Cmd
- Debug Azure CLI reference commands
- Get properties from a local JSON file
- Delete multiple resources using a script

There are also several reference commands that provide interactive prompts to help you learn the Azure CLI.

[\[+\] Expand table](#)

Command	Description	Example
<code>az interactive</code>	In a PowerShell console, get auto-completion, command descriptions, and examples as you type.	See Work in interactive mode for more information. Example: Type <code>az interactive</code> in your console window then <code>Enter</code> .
<code>az scenario</code>	Get end to end (E2E) examples and execute them with walk-through prompts.	Type <code>az scenario guide "keywords of interest"</code> in your console then <code>Enter</code> . The Azure CLI returns scenarios based on

Command	Description	Example
		your keyword(s). Example: az scenario guide "virtual machines"
az next	Get recommendations for the next Azure CLI command based on customer execution patterns. You are prompted to select a command path.	Type az next in your console then Enter. Pick from 1 of 7 preset command paths.
az find	This is an AI robot that will return reference examples for keywords.	Type az find "active subscription" in your console then Enter.

Another great way to learn to use the Azure CLI is through [Microsoft Learn Modules for CLI tools](#). When the module provides a sandbox environment, use the sandbox to try new commands without incurring costs.

Common usage questions

[\[+\] Expand table](#)

Question	Reference command answer	Learn more
How do I know what subscription I'm using?	az account show --output table	How to manage Azure subscriptions with the Azure CLI
How do I change my subscription?	az account set --subscription "mySubscriptionName"	How to manage Azure subscriptions with the Azure CLI
How can I set a default resource group and location?	az config set defaults.location=westus2 defaults.group=MyResourceGroup	Configure settings using az config
How do I query command output?	az account show --query "{tenantId:tenantId,subscriptionid:id}"	See How to query Azure CLI command output using a JMESPath query for many --query examples in Bash, PowerShell and Cmd.
How do I format output from a command?	az account list --output table	Output formats for Azure CLI commands

Question	Reference command answer	Learn more
Can you help me to learn Bash with Azure CLI commands?	Absolutely! We all have different technical strengths, and for that reason, Microsoft offers a How-to guide for learning Bash. Enjoy!	Learn to use Bash with the Azure CLI

Overview: Azure CLI terminology and support levels

Article • 05/21/2024

This article explains Azure CLI terminologies. There are syntax components, reference types, and statuses.

Azure CLI syntax components

The Azure CLI syntax is a combination of groups, references, commands, and parameters. Often the **full reference command** is referred to as **command**.

[+] Expand table

Azure Service	Reference group	Reference subgroup(s)	Command	Full reference command	Parameter Examples
Azure CLI	az config			<code>az config</code>	--local, --output -o
Azure Network	az network	application-gateway	create	<code>az network application-gateway create</code>	--name, --resource-group, --capacity
Azure DevOps	az pipelines	agent	list	<code>az pipelines agent list</code>	--pool-id, --agent-name, --demands

A **reference subgroup** can have multiple levels such as `az network application-gateway private-link ip-config add`

[+] Expand table

Reference Group	Subgroup 1	Subgroup 2	Subgroup 3	Command
network	application-gateway	private-link	ip-config	add

See [Reference list A to Z](#) for a complete list of reference commands.

What is reference type?

Azure CLI commands are either part of the **core** Azure CLI service, or they're an **extension**. Extensions are optional add-ons. The reference type determines the release schedule, status and installation method as described here:

[\[+\] Expand table](#)

Term	Core	Extension
References	Are part of the primary Azure CLI service	Are optional reference commands that must be installed
Install	Jointly with the MSI installer	Individually with az extension add
Released	On a schedule	As new features or updates become available
Status	Can be GA (Generally Available), preview or deprecated	Also can be GA, preview or deprecated

To get a list of command groups, run `az`. For a list of extensions, use [az extension list-available --output table](#) commands.

```
Azure CLI

# Get list of all command groups
az

# Get list of extensions
az extension list-available --output table
```

Core

Azure CLI references that have been published as a permanent part of the CLI are called **core references**. All core references install with the Azure CLI and you can't choose a subset of references. If you run the CLI through Azure Cloud Shell, core references are always up to date.

Extension

Extensions aren't shipped as part of the CLI but run as CLI commands. Some extensions are a permanent part of the Azure CLI, but often, an extension gives you access to preview and experimental commands. A single reference group, such as `az iot hub`, can have both core and extension commands. Here are two examples:

[+] Expand table

Full reference command	Is Core	Is Extension
az iot hub list	yes	
az iot hub job list		yes

You're prompted to install an extension upon first use. You can also install an extension by running the [az extension add](#) command.

You can learn more about extension references including installation and updating in [Use extensions with the Azure CLI](#). See [Available extensions for the Azure CLI](#) for a complete list of extension reference commands.

What is reference status?

Regardless of reference type, Azure CLI references fall into three status categories: **GA** (Generally Available), **public preview** or **deprecated**. It's the reference command status (not type) that determines stability.

[+] Expand table

	GA	Public preview	Deprecated
Stability	Permanent	Can change in response to customer feedback. Is subject to the terms of Microsoft Azure Previews .	Will be removed.

ⓘ Note

Warnings indicating **public preview** or **deprecated** are part of the Azure CLI command output and should be expected.

Most commands and parameters for a single reference have a single status, but not always. A GA reference that is being built out to offer more commands can have both GA and preview reference commands. As new parameters are added to increase functionality, a single command can also have parameters that fall under different status categories. Here are example references that have different statuses:

[+] Expand table

Full reference command	Parameters	Type	GA	Public preview	Deprecated
az network dns zone list	All	Core	yes		
az network dns zone create	--name, --resource-group, --if-none-match, --parent-name --existingParameter5	Core	yes		yes
	--newParameter6	Core	yes		
az network vhub list	All	Extension	yes		
az network vhub create	--address-prefix, --name, --resource-group, -vwan, --location, --sku --exsitingParameter7	Extension	yes		
	--newParameter8	Extension	yes		
az network firewall create	All	Extension	yes		

The above table is only an example and **isn't** representative of the current reference statuses for `az network`.

See also

- [Azure CLI support lifecycle](#)

Azure CLI support lifecycle

Article • 07/29/2024

Every Microsoft product has a lifecycle. The lifecycle begins when a product is released and ends when it's no longer supported. Knowing key dates in this lifecycle helps you make informed decisions about when to upgrade or make other changes to your software.

The Azure CLI support lifecycle is designed to provide clarity and predictability for customers regarding support for each release. The length of time and degree of support vary based on a few qualifications. The Azure CLI support lifecycle follows Microsoft's [Modern lifecycle policy](#) unless specified otherwise in this article.

Reference types

There are two Azure CLI reference types.

- **Core:** Azure CLI references that are published as a permanent part of the CLI are called "core" references. All core references install with the Azure CLI and you can't choose a subset of references. If you run the CLI through Azure Cloud Shell, core references are always up to date.
- **Extensions:** Extensions aren't shipped as part of the CLI but run as CLI commands. Some extensions are a permanent part of the Azure CLI, but often, an extension gives you access to more capabilities and preview features.

A single reference group can have both core and extension commands. For example, in the `az vm` reference group, `az vm nic add` is a core reference command while `az vm repair create` is an extension.

To learn more about Azure CLI extensions, see [Use and manage Azure CLI extensions](#).

Versioning

Core

Azure CLI core version is composed of three numbers: `major.minor.patch`.

- **Major versions:** Major changes, including architectural and fundamental design changes. We provide critical security fixes to the last minor version of the previous major version for three years.

- **Minor versions:** Updated for feature updates and improved Azure service coverage.
- **Patch versions:** Backward compatible fixes without new features or functionality changes.

Extensions

Azure CLI extensions follows [Semantic Versioning](#) for version numbering. Semantic versioning is a system that uses a three-part number format: `major.minor.patch`.

- **Major versions:** Breaking changes, often due to an incompatible API change.
- **Minor versions:** Backward compatible feature updates, or added functionality.
- **Patch versions:** Backward compatible fixes without new features or functionality changes.

Release cadence

The Azure CLI release cadence is designed to provide customers with predictable release dates, empowering businesses, developers, and the community to make strategic decisions when planning their roadmaps.

Azure CLI **Core** is updated monthly with a maximum of two releases introducing breaking changes. Breaking change releases occur during the second and fourth quarters of each year.

Azure CLI **extensions** don't have a specific release cadence and need to be upgraded manually.

Breaking changes might not affect existing scripts, but it's a good practice to verify your scripts in a test environment before performing an upgrade.

ⓘ Note

The release cadence might be slightly adjusted to be aligned with major announcements relevant to Azure.

Standard Term Support releases (STS)

The STS versions of Azure CLI core give you the most recent capabilities. STS releases don't experience breaking changes for six months. Breaking changes are released in the

second and fourth quarters of the calendar year, and usually align with Microsoft Build and Microsoft Ignite.

STS releases have the following characteristics:

- STS releases are released monthly and receive feature updates, bug fixes, and security fixes.
- For a STS version to be supported, you need to be on the latest minor version of the STS release.
- STS releases don't have a specific marker when running the `az version` command. This ensures backward compatibility with tools or scripts that might utilize the information.

Long Term Support releases (LTS)

The LTS versions of Azure CLI Core give you a "maintenance" period of 12 months following the "current" period. During this time, the LTS release doesn't experience breaking changes *unless required by a security fix*.

The LTS versions are based on Azure CLI releases that introduce major changes during the second quarter of the calendar year. This usually aligns with Microsoft Build.

LTS releases have the following characteristics:

- During the first six months, the LTS releases are released monthly and receive feature updates, bug fixes and security fixes.
- During the following 12 months, the LTS releases are released when necessary and only receive bug fixes and security fixes.
- For a LTS version to be supported, you need to be on the latest version of the LTS release.

Easily find the LTS release date by running the `az version` command. Notice the `azure-cli-support` line that indicates the LTS date.

```
Output
{
  "azure-cli": "2.48.0",
  "azure-cli-core": "2.48.0",
  "azure-cli-support": "LTS-20240521",
  "azure-cli-telemetry": "1.0.8",
  "extensions": {
  }
}
```

The following image illustrates the release cadence and support lifecycle of Azure CLI for STS and LTS releases.

	Q2 2024	Q4 2024	Q2 2025	Q4 2025	Q2 2026
STS	Current				
LTS		Current CLI v2.61 – v2.66	Maintenance CLI v2.66.x LTS		
STS			Current CLI v2.67 – v2.72		
LTS				Current CLI v2.73 – v2.78	Maintenance CLI v2.78.x LTS
STS					Current CLI v2.79 – 2.84

LTS: Long Term Support

STS: Standard Term Support

* Version numbers are subject to adjustments

Current:

- Versions receiving features updates, bug fixes and security fixes
- Released monthly

Maintenance:

- Versions receiving bug fixes and security fixes only
- Released as needed
- Will not introduce breaking changes unless related to a security issue

The following table provides examples of how the release cadence correlates with the support lifecycle:

ⓘ Note

The dates in the following table are for example purposes only. The actual release dates vary.

[\[\] Expand table](#)

Version number	Release schedule	Breaking changes	Release type	Support type	Supported until version	LTS version
2.61.0	May 2024	Yes	Major	STS	2.62.0	
2.62.0	June 2024		Minor	STS	2.63.0	
2.63.0	July 2024		Minor	STS	2.64.0	
2.64.0	August 2024		Minor	STS	2.65.0	
2.65.0	September 2024		Minor	STS	2.66.0	
2.66.0	October 2024		Minor	LTS	2.79.0	2.66.0

Version number	Release schedule	Breaking changes	Release type	Support type	Supported until version	LTS version
2.67.0	November 2024	Yes	Major	STS	2.68.0	2.66.x
2.68.0	December 2024		Minor	STS	2.69.0	2.66.x
2.69.0	January 2025		Minor	STS	2.70.0	2.66.x
2.70.0	February 2025		Minor	STS	2.71.0	2.66.x
2.71.0	March 2025		Minor	STS	2.72.0	2.66.x
2.72.0	April 2025		Minor	STS	2.73.0	2.66.x
2.73.0	May 2025	Yes	Major	STS	2.74.0	2.66.x
2.74.0	June 2025		Minor	STS	2.75.0	2.66.x
2.75.0	July 2025		Minor	STS	2.76.0	2.66.x
2.76.0	August 2025		Minor	STS	2.77.0	2.66.x
2.77.0	September 2025		Minor	STS	2.78.0	2.66.x
2.78.0	October 2025		Minor	STS	2.79.0	2.66.x
2.79.0	November 2025	Yes	Major	LTS	2.91.0	2.79.0
2.80.0	December 2025	Yes	Minor	STS	2.81.0	2.79.x

Examples:

- To qualify for support, you must update to 2.63.0 when released if you have Azure CLI 2.62.0 installed.
- If Azure CLI 2.66.0 is the last release of Azure CLI before breaking change in the fourth quarter of the calendar year, it's supported until the release of Azure CLI 2.79.0.

Breaking changes

Major version releases of Azure CLI introduce breaking changes and occur up to two times per year. Breaking changes can occur at any point for preview extensions. Non-GA extensions aren't required to adhere to breaking change policies.

For detailed information about breaking changes in Azure CLI, see [Release notes](#).

Supported environments

The Azure CLI runs on multiple operating systems (OS) and architectures. For a complete list, see [Install the Azure CLI](#). To be supported by Microsoft, the following criteria must be met:

- The version of Azure CLI is supported.
- The version of the OS is currently in mainstream support by the OS publisher.
- The dependencies required by the current version of Azure CLI are supported on the OS.

The Azure CLI ends support for a platform when one of the following conditions is met:

- The OS reaches its end of life as defined by the platform owner.
- The version of Python required by Azure CLI reaches its end of life, is no longer supported on the OS, or has an unfixed critical security issue.

Azure CLI announces the end of support of an OS or one of its dependencies within three months of the public announcement of the retirement.

Python dependency

Azure CLI depends on Python version 3.8 or above. The following table summarizes the expected end of support for each version of Python:

[+] [Expand table](#)

Python version	End of support date
3.8	October 2024
3.9	October 2025
3.10	October 2026
3.11	October 2027

Azure CLI can only be supported on operating systems where the above versions of Python are supported.

- **Windows:** The currently supported versions of Windows client and server meet Python version requirements.
- **macOS:** The currently supported version of macOS 12 and above meet Python version requirements.
- **Linux:**
 - Each supported operating system has a lifecycle defined by its sponsor organization.
 - Support is typically removed when an operating system goes out of mainline support, at which time we stop testing and supporting it.
 - Here are the supported Linux operating systems for the Azure CLI:

[\[+\] Expand table](#)

Operating system	Version	End of support
Ubuntu	20.04 LTS	April 2025
	22.04 LTS	April 2027
	24.04 LTS	June 2029
Debian	11	
	12	
Alpine	3.17	November 22, 2024
RHEL	8	May 31, 2029
	9	May 31, 2032
CentOS Stream	9	Estimated 2027
Mariner	2.0	

 **Note**

On RHEL 7, only Azure CLI 2.38 is supported. Microsoft is no longer providing updates or bug fixes for this distribution.

See also

- [Azure CLI terminology](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

-  [Open a documentation issue](#)
-  [Provide product feedback](#)

How to install the Azure CLI

Article • 06/19/2024

The Azure CLI is available to install in Windows, macOS and Linux environments. It can also be run in a Docker container and Azure Cloud Shell.

Install

The current version of the Azure CLI is **2.61.0**. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run [az version](#).

- [Install on Windows](#)
- [Install on macOS](#)
- Install on Linux or Windows Subsystem for Linux (WSL) ([What is WSL?](#))
 - [Install on RHEL/CentOS Stream with dnf](#)
 - [Install on SLES/OpenSUSE with zypper](#)
 - [Install on Ubuntu/Debian with apt](#)
 - [Install on Azure Linux with tdnf](#)
 - [Install from script](#)
- [Run in Docker container](#)
- [Run in Azure Cloud Shell](#)

Note

If you're using the Azure classic deployment model, [install the Azure classic CLI](#).

FAQ

Where is the Azure CLI installed?

When installing the Azure CLI, you can't select an install location. In Windows, the 32-bit Azure CLI installs in `C:\Program Files (x86)\Microsoft SDKs\Azure\CLI2` and the 64-bit in `C:\Program Files\Microsoft SDKs\Azure\CLI2`. In Linux, the Azure CLI is installed in

`/opt/az/` on Ubuntu and Debian, and in `/lib64/az/` on CentOS Stream, RHEL and Azure Linux.

User-specific configuration files are located in `$HOME/.azure` on macOS and Linux, and `%USERPROFILE%\azure` on Windows. These locations are known as the `AZURE_CONFIG_DIR`.

What version of the Azure CLI is installed?

Type `az version` in a terminal window to know what version of the Azure CLI is installed. Your output looks like this:

```
Output

{
  "azure-cli": "x.xx.0x",
  "azure-cli-core": "x.xx.x",
  "azure-cli-telemetry": "x.x.x",
  "extensions": {}
}
```

What extensions are installed?

Use the `az extension list` command to see installed extension. You can also use `az version`, but `az extension list` provides additional information including the installation path and status. For information on managing extensions, see [Use and manage extensions with the Azure CLI](#).

See also

- [Sign in with the Azure CLI](#)
- [Azure CLI Onboarding cheat sheet](#)
- [Find Azure CLI samples and published docs](#)
- [How to use the Azure CLI successfully](#)

 Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

more information, see [our contributor guide](#).

 [Provide product feedback](#)

Install Azure CLI on Windows

Article • 08/19/2024

The Azure Command-Line Interface (CLI) is a cross-platform command-line tool that can be installed locally on Windows computers. You can use the Azure CLI for Windows to connect to Azure and execute administrative commands on Azure resources. The Azure CLI for Windows can also be used from a browser through the Azure Cloud Shell or run from inside a Docker container.

For Windows, the Azure CLI is installed via an MSI or a ZIP package, which gives you access to the CLI through the Windows Command Prompt (CMD) or PowerShell. When you perform an installation for Windows Subsystem for Linux (WSL), packages are available for your Linux distribution. See the [main install page](#) for the list of supported package managers or how to install manually under WSL.

The current version of the Azure CLI is **2.64.0**. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run `az version`.

Install or update

The MSI and ZIP distributable are used for installing or updating the Azure CLI on Windows. You don't need to uninstall current versions before using the MSI installer because the MSI updates any existing version.

ⓘ Important

After the installation is complete, you will need to **close and reopen any active terminal window to use the Azure CLI**.

Microsoft Installer (MSI)

Latest version

Download and install the latest release of the Azure CLI. When the installer asks if it can make changes to your computer, select the "Yes" box.

[Latest MSI of the Azure CLI \(32-bit\)](#)

Latest MSI of the Azure CLI (64-bit)

If you have previously installed the Azure CLI, running either the 32-bit or 64-bit MSI will overwrite an existing installation.

Specific version

If you prefer, you can download a specific version of the Azure CLI by using a URL.

To download the MSI installer for a specific version, change the version segment in URL `https://azcliprod.blob.core.windows.net/msi/azure-cli-<version>.msi` (32-bit) or `https://azcliprod.blob.core.windows.net/msi/azure-cli-<version>-x64.msi` (64-bit).

For example, to install the 32-bit MSI of Azure CLI version [2.51.0](#), your URL would be `https://azcliprod.blob.core.windows.net/msi/azure-cli-2.51.0.msi`. The corresponding 64-bit install would be

`https://azcliprod.blob.core.windows.net/msi/azure-cli-2.51.0-x64.msi`.

Available Azure CLI versions can be found at [Azure CLI release notes](#). The 64-bit MSI is available from version [2.51.0](#).

Run the Azure CLI

You can now run the Azure CLI with the `az` command from either Windows Command Prompt or PowerShell.

Troubleshooting installation

Here are some common problems seen when installing the Azure CLI on Windows. If you experience a problem not covered here, [file an issue on GitHub](#).

PATH variable not set

The most common cause of this problem is that the active terminal window has not been restarted after installation. Close and reopen any active terminal window.

Proxy blocks connection

If you can't download the MSI installer because your proxy is blocking the connection, make sure that you have your proxy properly configured. For Windows 10, these settings are managed in the `Settings > Network & Internet > Proxy` pane. Contact your system administrator for the required settings, or for situations where your machine may be configuration-managed or require advanced setup.

Important

These settings are also required to be able to access Azure services with the CLI, from both PowerShell or the Command Prompt. In PowerShell, you do this with the following command:

PowerShell

```
(New-Object System.Net.WebClient).Proxy.Credentials = `  
[System.Net.CredentialCache]::DefaultNetworkCredentials
```

In order to get the MSI, your proxy needs to allow HTTPS connections to the following addresses:

- `https://aka.ms/`
- `https://azcliprod.blob.core.windows.net/`

For more information, see [Work behind a proxy](#) in the Azure CLI troubleshooting guide.

Slow response times

See [Migrate to 64-bit Azure CLI](#)

Enable Tab Completion in PowerShell

Tab completion, also known as "Azure CLI completers", provides completion on inputs to provide hints, enable discovery and speed up input entry. Command names, command group names, parameters and certain parameter values can be automatically inserted into the command line by pressing the `Tab` key.

Tab completion is enabled by default in Azure Cloud Shell and in most Linux distributions. Starting in Azure CLI version 2.49, you can enable tab completion for the Azure CLI in PowerShell. Follow these steps:

1. Create or edit the profile stored in the variable `$PROFILE`. The simplest way is to run `notepad $PROFILE` in PowerShell. For more information, see [How to create your profile](#) and [Profiles and execution policy](#).
2. Add the following code to your PowerShell profile:

```
PowerShell

Register-ArgumentCompleter -Native -CommandName az -ScriptBlock {
    param($commandName, $wordToComplete, $cursorPosition)
    $completion_file = New-TemporaryFile
    $env:ARGCOMPLETE_USE_TEMPFILES = 1
    $env:_ARGCOMPLETE_STDOUT_FILENAME = $completion_file
    $env:COMP_LINE = $wordToComplete
    $env:COMP_POINT = $cursorPosition
    $env:_ARGCOMPLETE = 1
    $env:_ARGCOMPLETE_SUPPRESS_SPACE = 0
    $env:_ARGCOMPLETE_IFS = "`n"
    $env:_ARGCOMPLETE_SHELL = 'powershell'
    az 2>&1 | Out-Null
    Get-Content $completion_file | Sort-Object | ForEach-Object {
        [System.Management.Automation.CompletionResult]::new($_, $_,
        "ParameterValue", $_)
    }
    Remove-Item $completion_file, Env:_ARGCOMPLETE_STDOUT_FILENAME,
Env:\ARGCOMPLETE_USE_TEMPFILES, Env:\COMP_LINE, Env:\COMP_POINT,
Env:\_ARGCOMPLETE, Env:\_ARGCOMPLETE_SUPPRESS_SPACE,
Env:\_ARGCOMPLETE_IFS, Env:\_ARGCOMPLETE_SHELL
}
```

3. To display all available options in the menu, add `Set-PSReadlineKeyHandler -Key Tab -Function MenuComplete` to your PowerShell profile.

Update the Azure CLI

Beginning with version [2.11.0](#), the Azure CLI provides an in-tool command to update to the latest version.

```
Azure CLI

az upgrade
```

This command also updates all installed extensions by default. For more `az upgrade` options, see the [command reference page](#). For Azure CLI versions prior to [2.11.0](#), update by reinstalling as described in [Install the Azure CLI](#).

If you are using ZIP distribution, please delete the old installation folder and extract the new version to the *same location*.

Migrate to 64-bit Azure CLI

Starting from 2.51.0, Azure CLI also provides 64-bit MSI which is recommended for better performance.

Follow these steps to migrate to Azure CLI 64-bit:

1. Check your current CLI version and installed extensions by running `az --version`.
2. Extensions will need to be reinstalled. It is recommended to perform a backup of the current extension folder `%userprofile%\azure\cliextensions` by renaming it in case you choose to revert back to 32-bit. This folder is created automatically when you reinstall an extension.
3. Download and install latest 64-bit installer as described in [Install or update](#). The 32-bit MSI will be automatically uninstalled.
4. Install extensions by running `az extension add --name <extension> --version <version>`. If you don't want to reinstall extensions manually, the Azure CLI will prompt you to install an extension on first use. For more information on installing extensions, see [How to install extensions](#).

If you have issues after migration, you can uninstall the 64-bit and reinstall the 32-bit MSI. If you have made a backup of your 32-bit extension folder, restore (rename) your extension folder after the change.

Uninstall

If you decide to uninstall the Azure CLI, we're sorry to see you go. Before you uninstall, use the `az feedback` command to let us know what could be improved or fixed. Our goal is to make the Azure CLI bug-free and user-friendly. If you found a bug, we'd appreciate it if you [file a GitHub issue](#).

You uninstall the Azure CLI from the Windows "Apps and Features" list. To uninstall:

[] [Expand table](#)

Platform	Instructions
Windows 11	Start > Settings > Apps > Installed apps
Windows 10	Start > Settings > Apps > Apps & Features

Platform	Instructions
Windows 8 and Windows 7	Start > Control Panel > Programs > Uninstall a program

Once on this screen type **Azure CLI** into the program search bar. The program to uninstall is listed as **Microsoft CLI 2.0 for Azure**. Select this application, then select the **Uninstall** button.

Remove data

If you don't plan to reinstall Azure CLI, remove its data from `C:\Users\<username>\.azure\msal_token_cache.bin` or `C:\Users\<username>\.azure\msal_token_cache.json`.

Next Steps

Now that you've installed the Azure CLI on Windows, learn about the different ways to sign in.

[Sign in with Azure CLI](#)

Install Azure CLI on macOS

Article • 10/08/2024

The Azure Command-Line Interface (CLI) is a cross-platform command-line tool that can be installed locally on macOS computers. You can use the Azure CLI to connect to Azure and execute administrative commands on Azure resources using interactive command-line prompts or a script.

For the macOS platform, install the Azure CLI with [homebrew package manager](#).

Homebrew makes it easy to keep your installation of the CLI update to date. The CLI package has been tested on macOS versions 10.9 and later.

The current version of the Azure CLI is **2.65.0**. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run `az version`.

Install with Homebrew

Homebrew is the easiest way to manage your CLI install. It provides convenient ways to install, update, and uninstall. If you don't have homebrew available on your system, [install homebrew](#) before continuing.

You can install the Azure CLI on macOS by updating your brew repository information, and then running the `install` command:

Bash

```
brew update && brew install azure-cli
```

Troubleshooting

If you encounter a problem when installing the CLI through Homebrew, here are some common errors. If you experience a problem not covered here, [file an issue on GitHub](#).

Completion isn't working

The Homebrew formula of Azure CLI installs a completion file named `az` in the Homebrew-managed completions directory (default location is

`/usr/local/etc/bash_completion.d/`). To enable completion, follow Homebrew's instructions [here ↗](#).

For Zsh, add the following two lines to the bottom of your `.zshrc` file, then save and reload your Zsh profile.

```
autoload bashcompinit && bashcompinit  
source $(brew --prefix)/etc/bash_completion.d/az
```

Unable to find Python or installed packages

There may be a minor version mismatch or other issue during homebrew installation. The CLI doesn't use a Python virtual environment, so it relies on finding the installed Python version. A possible fix is to install and relink the `python@3.10` dependency from Homebrew.

Bash

```
brew update && brew install python@3.10 && brew upgrade python@3.10  
brew link --overwrite python@3.10
```

CLI version 1.x is installed

If an out-of-date version was installed, it could be because of a stale homebrew cache. Follow the [update](#) instructions.

Proxy blocks connection

You may be unable to get resources from Homebrew unless you have correctly configured it to use your proxy. Follow the [Homebrew proxy configuration instructions ↗](#).

ⓘ Important

If you are behind a proxy, `HTTP_PROXY` and `HTTPS_PROXY` must be set to connect to Azure services with the CLI. If you are not using basic auth, it's recommended to export these variables in your `.bashrc` file. Always follow your business' security policies and the requirements of your system administrator.

In order to get the bottle resources from Homebrew, your proxy needs to allow HTTPS connections to the following addresses:

- `https://formulae.brew.sh`
- `https://homebrew.bintray.com`

Update

The CLI is regularly updated with bug fixes, improvements, new features, and preview functionality. A new release is available roughly every three weeks.

Beginning with version [2.11.0](#), the Azure CLI provides an in-tool command to update to the latest version.

```
Azure CLI  
az upgrade
```

This command also updates all installed extensions by default. For more `az upgrade` options, see the [command reference page](#). For Azure CLI versions prior to [2.11.0](#), update by reinstalling as described in [Install the Azure CLI](#).

You can also update your local Homebrew repository information and then upgrade the `azure-cli` package.

```
Bash  
brew update && brew upgrade azure-cli
```

Uninstall

If you decide to uninstall the Azure CLI, we're sorry to see you go. Before you uninstall, use the `az feedback` command to let us know what could be improved or fixed. Our goal is to make the Azure CLI bug-free and user-friendly. If you found a bug, we'd appreciate it if you [file a GitHub issue ↗](#).

Use homebrew to uninstall the `azure-cli` package.

```
Bash  
brew uninstall azure-cli
```

Remove data

If you don't plan to reinstall Azure CLI, remove its data.

```
Bash
```

```
rm -rf ~/.azure
```

Other installation methods

If you can't use homebrew to install the Azure CLI in your environment, it's possible to use the manual instructions for Linux. This process isn't officially maintained to be compatible with macOS. Using a package manager such as Homebrew is always recommended. Only use the manual installation method if you have no other option available.

For the manual installation instructions, see [Install Azure CLI on Linux manually](#).

Next Steps

Now that you've installed the Azure CLI on macOS, take a short tour of its features and common commands.

[Get started with the Azure CLI](#)

Install the Azure CLI on Linux

Article • 08/08/2023

This article explains how to install the Azure CLI on Linux. Select the appropriate package manager for your distribution from the options at the top of the page. Using a Linux distribution's package manager is recommended, but you may manually install the Azure CLI on Linux by selecting the [Install script](#) option.

The current version of the Azure CLI is 2.51.0. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run `az version`.

Before you begin

- The `apt` package manager contains x86_64 and ARM64 packages for the Azure CLI that has been tested on the following distributions.

Distribution	Version
Ubuntu	20.04 LTS (Focal Fossa), 22.04 (Jammy Jellyfish)
Debian	10 (Buster), 11 (Bullseye), 12 (Bookworm)

- Ubuntu 20.04 (Focal Fossa) and 20.10 (Groovy Gorilla) include an `azure-cli` package with version 2.0.81 provided by the `universe` repository. This package is outdated and not recommended. If this package is installed, remove the package before continuing by running the command `sudo apt remove azure-cli -y && sudo apt autoremove -y`. For more information on `apt remove`, see the [Ubuntu package management](#) or [ask ubuntu](#).

Install Azure CLI

There are two options to install the Azure CLI on your system. You can download an install script that runs the install commands for you, or you can execute the install commands yourself in a step-by-step process.

Option 1: Install with one command

The easiest way to install the Azure CLI is through a script maintained by the Azure CLI team. This script runs all installation commands in one step. This script is downloaded

via `curl` and piped directly to `bash` to install the CLI.

If you wish to inspect the contents of the script yourself before executing, download the script first using `curl` and inspect it in your favorite text editor.

Bash

```
curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash
```

Option 2: Step-by-step installation instructions

If you prefer a step-by-step installation process, complete the following steps to install the Azure CLI.

1. Get packages needed for the installation process:

Bash

```
sudo apt-get update
sudo apt-get install ca-certificates curl apt-transport-https lsb-release gnupg
```

2. Download and install the Microsoft signing key:

Bash

```
sudo mkdir -p /etc/apt/keyrings
curl -sLS https://packages.microsoft.com/keys/microsoft.asc |
    gpg --dearmor |
    sudo tee /etc/apt/keyrings/microsoft.gpg > /dev/null
sudo chmod go+r /etc/apt/keyrings/microsoft.gpg
```

3. Add the Azure CLI software repository:

Bash

```
AZ_REPO=$(lsb_release -cs)
echo "deb [arch=`dpkg --print-architecture` signed-by=/etc/apt/keyrings/microsoft.gpg]
https://packages.microsoft.com/repos/azure-cli/ $AZ_REPO main" |
    sudo tee /etc/apt/sources.list.d/azure-cli.list
```

4. Update repository information and install the `azure-cli` package:

Bash

```
sudo apt-get update  
sudo apt-get install azure-cli
```

Install specific version

Configure the `azure-cli` repository information as shown previously. Available versions can be found at [Azure CLI release notes](#).

1. To view available versions with command:

```
Bash  
  
apt-cache policy azure-cli
```

2. To install specific version:

```
Bash  
  
sudo apt-get install azure-cli=<version>-1~bullseye
```

Update Azure CLI

Beginning with version [2.11.0](#), the Azure CLI provides an in-tool command to update to the latest version.

```
Azure CLI  
  
az upgrade
```

This command also updates all installed extensions by default. For more `az upgrade` options, see the [command reference page](#). For Azure CLI versions prior to [2.11.0](#), update by reinstalling as described in [Install the Azure CLI](#).

You can also use `apt-get upgrade` to update the CLI package. This command upgrades all of the installed packages on your system that haven't had a dependency change.

```
Bash  
  
sudo apt-get update && sudo apt-get upgrade
```

To upgrade the CLI only, use `apt-get install`.

Bash

```
sudo apt-get update && sudo apt-get install --only-upgrade -y azure-cli
```

Uninstall Azure CLI

If you decide to uninstall the Azure CLI, we're sorry to see you go. Before you uninstall, use the `az feedback` command to let us know what could be improved or fixed. Our goal is to make the Azure CLI bug-free and user-friendly. If you found a bug, we'd appreciate it if you [file a GitHub issue](#).

1. Uninstall with `apt-get remove`:

Bash

```
sudo apt-get remove -y azure-cli
```

2. If you don't plan to reinstall the CLI, remove the Azure CLI repository information:

Bash

```
sudo rm /etc/apt/sources.list.d/azure-cli.list
```

3. If you aren't using other packages from Microsoft, remove the signing key:

Bash

```
sudo rm /etc/apt/trusted.gpg.d/microsoft.gpg
```

4. Remove any unneeded packages:

Bash

```
sudo apt autoremove
```

Remove data

If you don't plan to reinstall Azure CLI, remove its data.

Bash

```
rm -rf ~/.azure
```

Troubleshooting

Here are some common problems seen when installing with `apt`. If you experience a problem not covered here, [file an issue on GitHub](#).

The `azure-cli` package supports ARM64 architecture from CLI version [2.46.0](#).

No module issue on Ubuntu 20.04 (Focal)/WSL

If you installed `azure-cli` on `Focal` without adding the Azure CLI software repository in [step 3](#) of the manual install instructions or using our [script](#), you may encounter issues such as no module named 'decorator' or 'antlr4' as the package you installed is the outdated `azure-cli 2.0.81` from the `focal/universe` repository. Remove it first by running `sudo apt remove azure-cli -y && sudo apt autoremove -y`, then follow the above [instructions](#) to install the latest `azure-cli` package.

lsb_release doesn't return the correct base distribution version

Some Ubuntu or other Debian-derived distributions (such as Linux Mint) may not return the correct version name from `lsb_release`. This value is used in the install process to determine the package to install. If you know the code name of the Ubuntu or Debian version that your distribution is derived from, you can set the `AZ_REPO` value manually when [adding the repository](#). Otherwise, look up information for your distribution on how to determine the base distribution code name and set `AZ_REPO` to the correct value.

No package for your distribution

Sometimes, an updated Azure CLI package isn't immediately available following a distribution version release. The Azure CLI is designed to be resilient to future versions of dependencies and relies on as few of them as possible. If there's no package available for your base distribution, try a package for an earlier distribution.

To try a package for an earlier distribution, set the value of `AZ_REPO` manually when [adding the repository](#). For Ubuntu distributions, use the latest `jammy` repository:

Bash

```
AZ_REPO="jammy"
```

For Debian distributions, use the latest `bookworm` repository:

Bash

```
AZ_REPO="bookworm"
```

ⓘ Important

Distributions released before Ubuntu Bionic and Debian Buster are not supported.

Elementary OS (EOS) fails to install the Azure CLI

EOS fails to install the Azure CLI because `lsb_release` returns `HERA`, which is the EOS release name. The solution is to fix the file `/etc/apt/sources.list.d/azure-cli.list` and change `hera main` to `bionic main`.

Original file contents:

```
deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ hera main
```

Modified file contents

```
deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ bionic main
```

Proxy blocks connection

If you're unable to connect to an external resource due to a proxy, make sure that you've correctly set the `HTTP_PROXY` and `HTTPS_PROXY` variables in your shell. You need to contact your system administrator to know what host(s) and port(s) to use for these proxies.

The values in these variables are used by many Linux programs, and during the install process. To set these values:

Bash

```
# No auth
export HTTP_PROXY=http://[proxy]:[port]
export HTTPS_PROXY=https://[proxy]:[port]

# Basic auth
export HTTP_PROXY=http://[username]:[password]@[proxy]:[port]
export HTTPS_PROXY=https://[username]:[password]@[proxy]:[port]
```

ⓘ Important

If you are behind a proxy, these shell variables must be set to connect to Azure services with the CLI. If you are not using basic auth, it's recommended to export these variables in your `.bashrc` file. Always follow your business' security policies and the requirements of your system administrator.

You may also want to explicitly configure `apt` to always use this proxy. Make sure that the following lines appear in an `apt` configuration file in `/etc/apt/apt.conf.d/`. We recommend using either your existing global configuration file, an existing proxy configuration file, `40proxies`, or `99local`, but follow your system administration requirements.

apt.conf

```
Acquire {
    http::proxy "http://[username]:[password]@[proxy]:[port]";
    https::proxy "https://[username]:[password]@[proxy]:[port]";
}
```

If your proxy doesn't use basic auth, remove the `[username]:[password]@` portion of the proxy URI. If you require more information for proxy configuration, see the official Ubuntu documentation:

- [apt.conf manpage ↗](#)
- [Ubuntu wiki - apt-get howto ↗](#)

In order to get the Microsoft signing key and get the package from our repository, your proxy needs to allow HTTPS connections to the following address:

- <https://packages.microsoft.com>

CLI fails to install or run on Windows Subsystem for Linux

Since [Windows Subsystem for Linux \(WSL\)](#) is a system call translation layer on top of the Windows platform, you might experience an error when trying to install or run the Azure CLI. The CLI relies on some features that may have a bug in WSL. If you experience an error no matter how you install the CLI, there's a good chance it's an issue with WSL and not with the CLI install process.

To troubleshoot your WSL installation and possibly resolve issues:

- If you can, run an identical install process on a Linux machine or VM to see if it succeeds. If it does, your issue is likely related to WSL. To start a Linux VM in Azure, see the [create a Linux VM in the Azure portal](#) documentation.
- Make sure that you're running the latest version of WSL. To get the latest version, [update your Windows 10 installation ↗](#).
- Check for any [open issues ↗](#) with WSL that might address your problem. Often there are suggestions on how to work around the problem, or information about a release where the issue is fixed.
- If there are no existing issues for your problem, [file a new issue with WSL ↗](#) and make sure that you include as much information as possible.

If you continue to have issues installing or running on WSL, consider [installing the CLI for Windows](#).

Next Steps

Now that you've installed the Azure CLI, take a short tour of its features and common commands.

[Get started with the Azure CLI](#)

How to run the Azure CLI in a Docker container

Article • 08/06/2024

You can use Docker to run a standalone Linux container with the Azure CLI preinstalled. Docker gets you started quickly with an isolated environment to run the CLI in. The image can also be used as a base for your own deployments.

⚠️ Warning

Azure CLI 2.63.0 (released in August 2024) is the last version supported on the Alpine docker image. Microsoft will not be providing additional updates or bug fixes for this image. Consider migrating to the Azure Linux based image for the Azure CLI, which is fully supported.

To disable warning messages, run `az config set core.only_show_errors=yes`. For more information on Azure CLI configuration settings, see [CLI configuration values and environment variables](#).

Start the Docker container with Azure CLI preinstalled

Open a command prompt and start the Docker container with Azure CLI preinstalled using the following command. When used in a pipeline, it is recommended to use a deterministic tag name (for example `cbl-mariner2.0`)

Bash

```
docker run -it mcr.microsoft.com/azure-cli:cbl-mariner2.0
```

ⓘ Note

If you want to pick up the SSH keys from your user environment, use `-v ${HOME}/.ssh:/root/.ssh` to mount your SSH keys in the environment.

Bash

```
docker run -it -v ${HOME}/.ssh:/root/.ssh mcr.microsoft.com/azure-
```

```
cli:cbl-mariner2.0
```

The CLI is installed on the image as the `az` command in `/usr/local/bin`.

Run the Docker container with a specific version of the Azure CLI

Available versions can be found at [Azure CLI release notes](#).

To run a specific version of the Azure CLI in the Docker container, use this command. Change `2.62.0` with your desired version.

```
Bash
```

```
docker run -it mcr.microsoft.com/azure-cli:2.62.0-cbl-mariner2.0
```

Update Docker image

Updating with Docker requires both pulling the new image and re-creating any existing containers. For this reason, you should try to avoid using a container that hosts the CLI as a data store.

Update your local image with `docker pull`.

```
Bash
```

```
docker pull mcr.microsoft.com/azure-cli:cbl-mariner2.0
```

Uninstall Docker image

If you decide to uninstall the Azure CLI, we're sorry to see you go. Before you uninstall, use the `az feedback` command to let us know what could be improved or fixed. Our goal is to make the Azure CLI bug-free and user-friendly. If you found a bug, we'd appreciate it if you [file a GitHub issue](#).

After halting any containers running the CLI image, remove it.

```
Bash
```

```
docker rmi mcr.microsoft.com/azure-cli
```

Next Steps

Now that you're ready to use the Azure CLI in a Docker container, take a short tour of its features and common commands.

[Get started with the Azure CLI](#)

 Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Get started with Azure Cloud Shell

Article • 01/22/2024

This document details how to get started using Azure Cloud Shell.

Prerequisites

Before you can use Azure Cloud Shell, you must register the **Microsoft.CloudShell** resource provider. Access to resources is enabled through provider namespaces that must be registered in your subscription. You only need to register the namespace once per subscription.

To see all resource providers, and the registration status for your subscription:

1. Sign in to the [Azure portal](#).
2. On the Azure portal menu, search for **Subscriptions**. Select it from the available options.
3. Select the subscription you want to view.
4. On the left menu, under **Settings**, select **Resource providers**.
5. In the search box, enter `cloudshell` to search for the resource provider.
6. Select the **Microsoft.CloudShell** resource provider register from the provider list.
7. Select **Register** to change the status from **unregistered** to **Registered**.

The screenshot shows the Microsoft Azure portal interface. The user is on the 'Resource providers' page under the 'Contoso (carolb)' subscription. The left sidebar is expanded, showing various navigation options like 'Programmatic deployment', 'Billing properties', and 'Resource providers'. The 'Resource providers' option is highlighted with a red box. The main content area shows a table with one row:

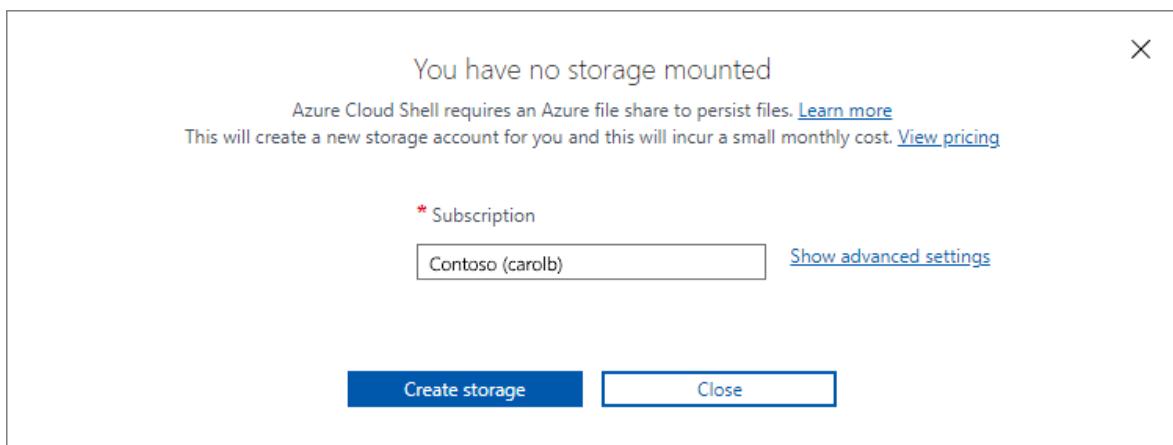
Provider	Status
Microsoft.CloudShell	x unregistered

Start Cloud Shell

1. Launch Cloud Shell from the top navigation of the Azure portal.



The first time you start Cloud Shell you're prompted to create an Azure Storage account for the Azure file share.



2. Select the **Subscription** used to create the storage account and file share.

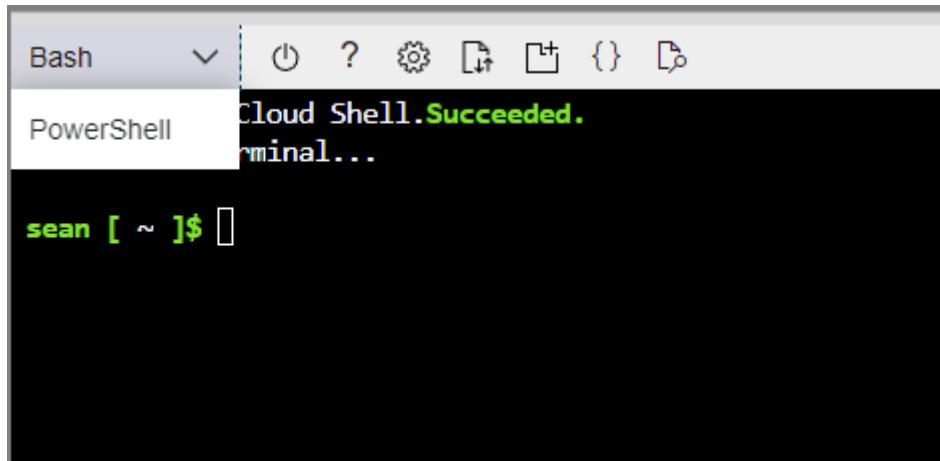
3. Select **Create storage**.

! Note

By following these steps, Cloud Shell creates a standard storage account and allocates 5 GB of storage for the file share. You can also create a storage account manually and specify the storage account and file share to use. If you use a Premium storage account, Cloud Shell allocates 100 GB of storage for the file share.

Select your shell environment

Cloud Shell allows you to select either **Bash** or **PowerShell** for your command-line experience.



Set your subscription

1. List subscriptions you have access to.

A screenshot of the Microsoft Azure Cloud Shell interface. The top bar shows "Azure CLI" as the selected environment. The main terminal window contains the command "az account list" in blue text, which is a command used to list all available Azure subscriptions. The terminal window has a light gray background and a dark gray border.

2. Set your preferred subscription:

A screenshot of the Microsoft Azure Cloud Shell interface. The top bar shows "Azure CLI" as the selected environment. The main terminal window contains the command "az account set --subscription 'my-subscription-name'" in blue text, which is a command used to set the current subscription for the Azure CLI. The terminal window has a light gray background and a dark gray border.

💡 Tip

Your subscription is remembered for future sessions using
`/home/<user>/azureProfile.json`.

Get a list of Azure commands

Azure CLI

Run the following command to see a list of all Azure CLI commands.

Azure CLI

`az`

Run the following command to get a list of Azure CLI commands that apply to WebApps:

Azure CLI

`az webapp --help`

Next steps

- [Learn about Azure PowerShell](#)
- [Learn about Azure CLI](#)
- [Learn about persisting files in Cloud Shell](#)
- [Learn about Azure Files storage](#)

How to update the Azure CLI

Article • 08/02/2023

You can rely on package managers to update a local install of the Azure CLI on Windows, macOS and Linux environments (see the `Update` section in each platform-specific install instruction). The CLI also provides in-tool commands to upgrade manually or automatically.

Manual Update

Beginning with version [2.11.0](#), the Azure CLI provides an in-tool command to update to the latest version.

```
Azure CLI
az upgrade
```

This command also updates all installed extensions by default. For more `az upgrade` options, see the [command reference page](#). For Azure CLI versions prior to [2.11.0](#), update by reinstalling as described in [Install the Azure CLI](#).

`az upgrade` is supported on Windows, macOS and some Linux distros as long as installation is supported. It only supports upgrading to the latest version. If you're running the Azure CLI through Azure Cloud Shell, you're most likely already using the most recent Azure CLI install. If not due to cases like ad-hoc release of a minor bug fix version, you need to wait for the next build of Azure Cloud Shell as `az upgrade` isn't supported in Azure Cloud Shell.

When `azure-cli` is already the latest version, running `az upgrade` checks and updates all installed [extensions](#).

Automatic Update

By default, autoupgrade for Azure CLI is disabled. If you would like to keep up with the latest version, you can enable autoupgrade through [configuration](#).

```
Azure CLI
az config set auto-upgrade.enable=yes
```

The Azure CLI will check new versions regularly and prompt you to upgrade after any command finishes running once the update is available.

The prompt message and output messages during upgrade may interrupt your command result if it's assigned to some variable or in an automated flow. To avoid interruption, you can use the following configuration to allow the update to happen automatically without confirmation, and only show warnings and errors during the upgrade.

```
Azure CLI
```

```
az config set auto-upgrade.prompt=no
```

By default, all installed extensions are also updated. You can disable extension update through configuration.

```
Azure CLI
```

```
az config set auto-upgrade.all=no
```

ⓘ Note

Please wait for `az upgrade` to complete before proceeding to the next set of commands, else the new versions of the CLI (+extensions) may have breaking changes.

If you decide not to use the automatic update feature anymore for cases like keeping command scripts running stably, you can turn it off through configuration.

```
Azure CLI
```

```
az config set auto-upgrade.enable=no
```

Azure CLI endpoints for proxy bypass

Article • 07/24/2024

If your organization is secured with a firewall or proxy server, you must add certain IP (internet protocol) addresses and domain URLs (uniform resource locators) to the **allowlist** prior to installing the Azure CLI.

Two URLs are used during installation to download Python packages: pypi.org and files.pythonhosted.org.

The following tables provide lists of the endpoints and suffixes used by the Azure CLI. These endpoints are specific to the Azure cloud where your organization is deployed. We don't recommend adding other Azure CLI-related URLs aside from the URLs needed for specific Azure CLI reference groups, although you may want to add URLs related to other Microsoft products and services.

Endpoints

Public Cloud

[Expand table](#)

Endpoint group	Endpoint
management	https://management.core.windows.net/
resource_manager	https://management.azure.com/
sql_management	https://management.core.windows.net:8443/
batch_resource_id	https://batch.core.windows.net/
gallery	https://gallery.azure.com/
active_directory	https://login.microsoftonline.com_
active_directory_resource_id	https://management.core.windows.net/
active_directory_graph_resource_id	https://graph.windows.net/
microsoft_graph_resource_id	https://graph.microsoft.com/
active_directory_data_lake_resource_id	https://datalake.azure.net/

Endpoint group	Endpoint
vm_image_alias_doc	https://raw.githubusercontent.com/Azure/azure-rest-api-specs/main/arm-compute/quickstart-templates/aliases.json
media_resource_id	https://rest.media.azure.net
osrdbms_resource_id	https://osrdbms-aad.database.windows.net
app_insights_resource_id	https://api.applicationinsights.io
log_analytics_resource_id	https://api.loganalytics.io
app_insights_telemetry_channel_resource_id	https://dc.applicationinsights.azure.com/v2/track
synapse_analytics_resource_id	https://dev.azuresynthesize.net
attestation_resource_id	https://attest.azure.net
portal	https://portal.azure.com

Endpoint suffixes

[Expand table](#)

Suffix name	Suffix
storage_endpoint	*.core.windows.net
storage_sync_endpoint	*.afs.azure.net
keyvault_dns	*.vault.azure.net
mhsm_dns	*.managedhsm.azure.net
sql_server_hostname	*.database.windows.net
mysql_server_endpoint	*.mysql.database.azure.com
postgresql_server_endpoint	*.postgres.database.azure.com
mariadb_server_endpoint	*.mariadb.database.azure.com
azure_datalake_store_file_system_endpoint	*.azuredatalakestore.net
azure_datalake_analytics_catalog_and_job_endpoint	*.azuredatalakeanalytics.net
acr_login_server_endpoint	*.azurecr.io

Suffix name	Suffix
synapse_analytics_endpoint	*.dev.azuresynthesize.net
attestation_endpoint	*.attest.azure.net

Extensions

Azure CLI extensions are optional and installed separately. The Azure CLI uses <https://aka.ms/azure-cli-extension-index-v1> to fetch a list of extensions. This *aka.ms* link points to https://github.com/Azure/azure-cli/blob/3feea02888ea67f033f407174a3a7a340158b81a/src/azure-cli-core/azure/cli/core/extension/_index.py#L11.

All extensions install with endpoint **azcliprod.blob.core.windows.net** with the following exceptions:

[] Expand table

Extension name	Endpoint
arcappliance	arcplatformcliextprod.blob.core.windows.net
customlocation	
arcdataproxy	azurearcdataproxy.blob.core.windows.net
azure-batch-cli-extensions	github.com\Azure
azure-devops	
azure-iot	
baremetal-infrastructure	
csvmware	
deploy-to-azure	
sap-hana	
azure-cli-ml	azurecliext.blob.core.windows.net
azurestackhci	hybridaksstorage.z13.web.core.windows.net
Hybridaks	
fzf	pahealyfzf.blob.core.windows.net

Extension name	Endpoint
image-copy-extension	files.pythonhosted.org
managementpartner	
resource-graph	
ml	azuremlsdktestpypi.blob.core.windows.net
qbs	qbsazcliextension.blob.core.windows.net

See also

- [Azure CLI troubleshooting - Work behind a proxy](#)
- [Work with existing on-premises proxy servers](#)
- [Azure Firewall threat intelligence configuration - Allowlist addresses](#)
- [Azure IP ranges and Service Tags](#)
 - [Public Cloud ↗](#)
 - [US Government Cloud ↗](#)
 - [China Cloud ↗](#)

Authenticate to Azure using Azure CLI

Article • 09/24/2024

The Azure CLI supports several authentication methods. Restrict sign-in permissions for your use case to keep your Azure resources secure.

Sign into Azure with Azure CLI

There are five authentication options when working with the Azure CLI:

[+] Expand table

Authentication method	Advantage
Azure Cloud Shell	Azure Cloud Shell automatically logs you in and is the easiest way to get started.
Sign in interactively	This is a good option when learning Azure CLI commands and running the Azure CLI locally. Login through your browser with the az login command. Interactive login also gives you a subscription selector to automatically set your default subscription.
Sign in with a managed identity	Managed identities provide an Azure-managed identity for applications to use when connecting to resources that support Microsoft Entra authentication. Using a managed identity eliminates the need for you to manage secrets, credentials, certificates, and keys.
Sign in using a service principal	When you write scripts, using a service principal is the recommended authentication approach. You grant just the appropriate permissions needed to a service principal keeping your automation secure.

Multi-factor authentication (MFA)

Microsoft announced in May, 2024, that it will require [MFA for all Azure users](#). This announcement was followed by a June 2024 [update](#) that provided more information on scope and timing.

MFA will only impact Microsoft Entra ID [users](#). It will not impact [service principals](#) or [managed identities](#).

Find or change your current subscription

After you sign in, CLI commands are run against your default subscription. If you have multiple subscriptions, change your default subscription using `az account set --subscription`.

Azure CLI

```
az account set --subscription "<subscription ID or name>"
```

To learn more about managing Azure subscriptions, see [How to manage Azure subscriptions with the Azure CLI](#).

Refresh tokens

When you sign in with a user account, Azure CLI generates and stores an authentication refresh token. Because access tokens are valid for only a short period of time, a refresh token is issued at the same time the access token is issued. The client application can then exchange this refresh token for a new access token when needed. For more information on token lifetime and expiration, see [Refresh tokens in the Microsoft identity platform](#).

Use the `az account get-access-token` command to retrieve the access token:

Azure CLI

```
# get access token for the active subscription
az account get-access-token

# get access token for a specific subscription
az account get-access-token --subscription "<subscription ID or name>"
```

Here is some additional information about access token expiration dates:

- Expiration dates are updated in a format that is supported by [MSAL-based Azure CLI](#).
- Starting from Azure CLI 2.54.0, `az account get-access-token` returns the `expires_on` property alongside the `expiresOn` property for the token expiration time.
- The `expires_on` property represents a Portable Operating System Interface (POSIX) timestamp while the `expiresOn` property represents a local datetime.
- The `expiresOn` property doesn't express "fold" when Daylight Saving Time ends. This can cause problems in countries or regions where Daylight Saving Time is

adopted. For more information on "fold", see [PEP 495 – Local Time Disambiguation](#).

- We recommend for downstream applications to use the `expires_on` property, because it uses the Universal Time Code (UTC).

Example output:

JSON

```
{  
    "accessToken": "...",  
    "expiresOn": "2023-10-31 21:59:10.000000",  
    "expires_on": 1698760750,  
    "subscription": "...",  
    "tenant": "...",  
    "tokenType": "Bearer"  
}
```

ⓘ Note

Depending on your sign in method, your tenant may have [Conditional Access policies](#) that restrict your access to certain resources.

See also

- [Azure CLI onboarding cheat sheet](#)
- [Manage Azure subscriptions with the Azure CLI](#)
- Find Azure CLI samples and [published articles](#)

Sign into Azure interactively using the Azure CLI

Article • 09/24/2024

Interactive logins to Azure offer a more intuitive and flexible user experience. Interactive login with Azure CLI allows users to authenticate to Azure directly through the [az login](#) command, which is useful for ad-hoc management tasks and for environments that require manual sign-in, such as those customers with multi-factor authentication (MFA). This method simplifies access for script testing, learning, and on-the-fly management without needing to preconfigure service principals or other noninteractive authentication methods.

Prerequisites

- [Install the Azure CLI](#)

Interactive login

To sign in interactively, use the [az login](#) command. Beginning with Azure CLI version [2.61.0](#), Azure CLI uses Web Account Manager (WAM) on Windows, and a browser-based login on Linux and macOS by default.

```
Azure CLI
az login
```

Subscription selector

Beginning with Azure CLI version [2.61.0](#), if you have access to multiple subscriptions, you're prompted to select an Azure subscription at time of login, as shown in the following example.

```
Output
Retrieving subscriptions for the selection...
[Tenant and subscription selection]
No      Subscription name          Subscription ID
Tenant name
```

```
-----  
-----  
[1] Facility Services Subscription      00000000-0000-0000-0000-  
000000000000 Contoso  
[2] Finance Department Subscription     00000000-0000-0000-0000-  
000000000000 Contoso  
[3] Human Resources Subscription        00000000-0000-0000-0000-  
000000000000 Contoso  
[4] * Information Technology Subscription 00000000-0000-0000-0000-  
000000000000 Contoso
```

The default is marked with an *; the default tenant is 'Contoso' and subscription is 'Information Technology Subscription' (00000000-0000-0000-0000-000000000000).

Select a subscription and tenant (Type a number or Enter for no changes): 2

Tenant: Contoso

Subscription: Finance Department Subscription (00000000-0000-0000-0000-000000000000)

[Announcements]

With the new Azure CLI login experience, you can select the subscription you want to use more easily.

Learn more about it and its configuration at
<https://go.microsoft.com/fwlink/?linkid=2271236>

If you encounter any problem, please open an issue at
<https://aka.ms/azclibug>

The next time you login, the previously selected tenant and subscription is marked as the default with an asterisk (*) next to its number. This allows you to press **Enter** to select the default subscription.

Commands run against the selected subscription by default. You can still use `az account set` to change your subscription from a command line at any time. For more information, see [How to manage Azure subscriptions with the Azure CLI](#).

Here are some guidelines about the subscription selector to keep in mind:

- The subscription selector is only available in 64-bit Windows, Linux, or macOS.
- The subscription selector is only available when using the `az login` command.
- You aren't prompted to select a subscription when you're logging in with a service principal or managed identity.

If want to disable the subscription selector feature, set the `core.login_experience_v2` configuration property to `off`.

```
Azure CLI
```

```
az config set core.login_experience_v2=off  
az login
```

Sign in with Web Account Manager (WAM) on Windows

Beginning with Azure CLI version [2.61.0](#), Web Account Manager (WAM) is now the default authentication method on Windows. WAM is a Windows 10+ component that acts as an authentication broker. (An authentication broker is an application that runs on a user's machine that manages the authentication handshakes and token maintenance for connected accounts.)

Using WAM has several benefits:

- Enhanced security. See [Conditional Access: Token protection \(preview\)](#).
- Support for Windows Hello, conditional access policies, and FIDO keys.
- Streamlined single sign-on.
- Bug fixes and enhancements shipped with Windows.

If you encounter an issue and want to revert to the previous browser-based authentication method, set the `core.enable_broker_on_windows` configuration property to `false`.

```
Azure CLI
```

```
az account clear  
az config set core.enable_broker_on_windows=false  
az login
```

WAM is available on Windows 10 and later, and on Windows Server 2019 and later.

Sign in with a browser

The Azure CLI defaults to a browser-based authentication method when one of the following is true:

- The operating system (OS) is Mac, or Linux, or the Windows OS is earlier than Windows 10 or Windows Server 2019.
- The `core.enable_broker_on_windows` configuration property is set to `false`.

Follow these steps to sign in with a browser:

1. Run the `az login` command.

```
Azure CLI
```

```
az login
```

If the Azure CLI can open your default browser, it initiates [authorization code flow](#) and opens the default browser to load an Azure sign-in page.

Otherwise, it initiates the [device code flow](#) and instructs you to open a browser page at <https://aka.ms/devicelogin>. Then, enter the code displayed in your terminal.

If no web browser is available or the web browser fails to open, you may force device code flow with `az login --use-device-code`.

2. Sign in with your account credentials in the browser.

Sign in with credentials on the command line

Provide your Azure user credentials on the command line. Only use this authentication method for learning Azure CLI commands. Production-level applications should use a service principal or managed identity.

This approach doesn't work with Microsoft accounts or accounts that have two-factor authentication enabled. You receive an *interactive authentication is needed* message.

```
Azure CLI
```

```
az login --user <username> --password <password>
```

ⓘ Important

If you want to avoid displaying your password on console and are using `az login` interactively, use the `read -s` command under `bash`.

```
Bash
```

```
read -sp "Azure password: " AZ_PASS && echo && az login -u <username> -p $AZ_PASS
```

Under PowerShell, use the `Get-Credential` cmdlet.

PowerShell

```
$AzCred = Get-Credential -UserName <username>
az login -u $AzCred.UserName -p $AzCred.GetNetworkCredential().Password
```

Sign in with a different tenant

You can select a tenant to sign in under with the `--tenant` argument. The value of this argument can either be an `.onmicrosoft.com` domain or the Azure object ID for the tenant. Both interactive and command-line sign-in methods work with `--tenant`.

In select environments and beginning in Azure CLI version [2.61.0](#), you need to first disable the subscription selector by setting the `core.login_experience_v2` configuration property to `off`.

Azure CLI

```
# disable the subscription selector (v. 2.61.0 and up)
az config set core.login_experience_v2=off

# login with a tenant ID
az login --tenant 00000000-0000-0000-0000-000000000000
```

To reenable the subscription selector, run `az config set core.login_experience_v2=on`. For more information on the subscription selector, see [Interactive login](#).

After signing in, if you want to change your active tenant, see [How-to change your active tenant](#).

Sign in using `--scope`

```
az login --scope https://management.core.windows.net//.default
```

Multi-factor authentication (MFA)

Microsoft announced in May, 2024, that it will require [MFA for all Azure users](#). This announcement was followed by a June 2024 [update](#) that provided more information on scope and timing.

MFA will only impact Microsoft Entra ID [users](#). It will not impact [service principals](#) or [managed identities](#).

Logout

To remove access to Azure, use the `az logout` command.

```
Azure CLI
```

```
az logout
```

Clear your subscription cache

To update your subscription list, use the `az account clear` command. You will need to sign in again to see an updated list.

```
Azure CLI
```

```
az account clear
```

```
az login
```

Clearing your subscription cache is not technically the same process as logging out of Azure. However, when you clear your subscription cache, you cannot run Azure CLI commands, including `az account set`, until you sign in again.

Refresh tokens

When you sign in with a user account, Azure CLI generates and stores an authentication refresh token. Because access tokens are valid for only a short period of time, a refresh token is issued at the same time the access token is issued. The client application can then exchange this refresh token for a new access token when needed. For more information on token lifetime and expiration, see [Refresh tokens in the Microsoft identity platform](#).

Use the `az account get-access-token` command to retrieve the access token:

Azure CLI

```
# get access token for the active subscription
az account get-access-token

# get access token for a specific subscription
az account get-access-token --subscription "<subscription ID or name>"
```

Here is some additional information about access token expiration dates:

- Expiration dates are updated in a format that is supported by [MSAL-based Azure CLI](#).
- Starting from Azure CLI 2.54.0, `az account get-access-token` returns the `expires_on` property alongside the `expiresOn` property for the token expiration time.
- The `expires_on` property represents a Portable Operating System Interface (POSIX) timestamp while the `expiresOn` property represents a local datetime.
- The `expiresOn` property doesn't express "fold" when Daylight Saving Time ends. This can cause problems in countries or regions where Daylight Saving Time is adopted. For more information on "fold", see [PEP 495 – Local Time Disambiguation](#).
- We recommend for downstream applications to use the `expires_on` property, because it uses the Universal Time Code (UTC).

Example output:

JSON

```
{
  "accessToken": "...",
  "expiresOn": "2023-10-31 21:59:10.000000",
  "expires_on": 1698760750,
  "subscription": "...",
  "tenant": "...",
  "tokenType": "Bearer"
}
```

Troubleshooting

When your default browser is Microsoft Edge, you might encounter the following error when attempting to sign in to Azure interactively with `az login`: *"The connection for this site isn't secure."* To resolve this issue, visit [edge://net-internals/#hsts](#) in Microsoft Edge. Add `localhost` under "Delete domain security policy" and select `Delete`.

See also

- [Azure CLI Onboarding cheat sheet](#)
- [Find Azure CLI samples and published docs](#)

Sign into Azure with a managed identity using the Azure CLI

Article • 09/02/2024

On resources configured for managed identities for Azure resources, you can sign in using the managed identity. Here are some of the benefits of using managed identities:

- You don't need to manage credentials. Credentials aren't even accessible to you.
- You can use managed identities to authenticate to any resource that supports Microsoft Entra authentication, including your own applications.
- Managed identities can be used at no extra cost.

Signing in with the resource's identity is done through the `--identity` flag.

To sign in with a system-assigned managed identity:

```
Azure CLI
```

```
az login --identity
```

To sign in with a user-assigned managed identity, you must specify the client ID, object ID or resource ID of the user-assigned managed identity with `--username`:

```
Azure CLI
```

```
az login --identity --username <client_id|object_id|resource_id>
```

To learn more about managed identities for Azure resources, see [What are managed identities for Azure resources?](#)

Here are three articles showing the use of the `az login --identity` command:

- [How to use managed identities for Azure resources on an Azure VM for sign-in](#).
- [Use an Azure managed identity to authenticate to an Azure container registry](#)
- [How to use managed identities with Azure container instances](#)

Here are popular articles for specific Azure services that show how to work with managed identities using Azure CLI commands:

- [Use a managed identity in Azure Kubernetes Service \(AKS\)](#)
- [How to use Azure managed identities for Azure Service and Azure functions](#)
- [Create an Azure SQL Managed Instance with a user-assigned managed identity](#)

- How to use Managed Identity with [Azure Communication Services](#)
- Configure managed identities with Microsoft Entra ID for your [Azure Cosmos DB](#) account

See also

- [A to Z list](#) of Azure CLI reference commands that manage Azure identities.

Sign into Azure with a service principal using the Azure CLI

Article • 09/02/2024

Service principals are accounts not tied to any particular user, which can have permissions on them assigned through predefined roles. Authenticating with a service principal is the best way to write secure scripts or programs, allowing you to apply both permissions restrictions and locally stored static credential information. To learn more about service principals, see [Work with Azure service principals using the Azure CLI](#).

To sign in with a service principal, you need:

- The URL or name associated with the service principal
- The service principal password, or the X509 certificate used to create the service principal in PEM format
- The tenant associated with the service principal, as either an `.onmicrosoft.com` domain or Azure object ID

Note two important facts when working with service principals and the Azure CLI:

- A **CERTIFICATE** must be appended to the **PRIVATE KEY** within a PEM file. For an example of a PEM file format, see [Certificate-based authentication](#).
- If your service principal uses a certificate that is stored in Key Vault, that certificate's private key must be available without signing in to Azure. To retrieve the certificate for `az login`, see [Retrieve certificate from Key Vault](#).

Azure CLI

```
az login --service-principal -u <app-id> -p <password-or-cert> --tenant <tenant>
```

ⓘ Important

If you want to avoid displaying your password on console and are using `az login` interactively, use the `read -s` command under `bash`.

Bash

```
read -sp "Azure password: " AZ_PASS && echo && az login --service-principal -u <app-id> -p $AZ_PASS --tenant <tenant>
```

Under PowerShell, use the `Get-Credential` cmdlet.

PowerShell

```
$AzCred = Get-Credential -UserName <app-id>
az login --service-principal -u $AzCred.UserName -p
$AzCred.GetNetworkCredential().Password --tenant <tenant>
```

See also

- [Azure CLI Onboarding cheat sheet](#)
- [Manage Azure subscriptions with the Azure CLI](#)
- [Create an Azure service principal using Azure CLI](#)
- [Find Azure CLI samples and published docs](#)

Azure CLI release highlights

Article • 09/24/2024

This page highlights new features and upcoming changes for the Azure CLI.

Multi-factor authentication (MFA)

Microsoft announced in May, 2024, that it will require [MFA for all Azure users](#). This announcement was followed by a June 2024 [update](#) that provided more information on scope and timing.

MFA will only impact Microsoft Entra ID [users](#). It will not impact [service principals](#) or [managed identities](#).

As more information becomes available that is specific to the Azure CLI, we'll post more details on this page.

Docker container image

Beginning with Azure CLI version 2.67.0, to be released on November 19, 2024, the default Docker container image for Azure CLI will be based on Azure Linux. To avoid any disruptions, review and update any dependencies you may have on the default Docker container image for Azure CLI.

For more information see [How to run the Azure CLI in a Docker container](#).

Select your subscription at time of login

We've listened to your feedback and improved the Azure CLI interactive login experience to include a subscription selector. To use the new feature, see [Sign in interactively with Azure CLI](#).

Protect sensitive information

Beginning in [Azure CLI 2.57](#), a warning message can be displayed when reference commands result in the output of sensitive information. For more information, see [Manage Azure secrets using the Azure CLI](#).

Azure Copilot for Azure CLI

[Microsoft Copilot for Azure](#) (preview) is published! Copilot is an AI-powered tool to help you do more with Azure. It unifies knowledge and data across hundreds of Azure services to increase productivity, reduce costs, and provide deep insights. Microsoft Copilot for Azure (preview) helps you learn about Azure by answering questions, and it provides information tailored to your own Azure resources and environment. By letting you express your goals in natural language, Copilot simplifies your Azure management experience. This benefits Azure CLI users because the knowledge of Azure CLI is built into Copilot!

Access Microsoft Copilot for Azure (preview) in the Azure portal, and tell Copilot what you would like to do using Azure CLI. For example:

- I want to create a virtual machine using Azure CLI.
- I want to update service principal credentials using Azure CLI.
- I want to create a web app using Azure CLI.

To enable access to Microsoft Copilot for Azure (preview) for your organization, [complete the registration form](#). The application process only needs to be completed once per tenant. Check with your administrator if you have questions about joining the preview.

Reduced Docker image size

With the release of Azure CLI version 2.54.0, the size of the Docker image of `azure-cli` is reduced from 1.1 GB to 700 MB! This reduction is a 36.3% decrease resulting in improved download speed and faster startup. For more information, see "Trim Azure CLI's docker image size" in [Azure Command-line Tools Ignite 2023 Announcement](#).

64-bit Windows install

You can now [install the Azure CLI on Windows](#) with a 64-bit MSI. The 32-bit MSI, PowerShell command, and Windows Package Manager are still available, but the 64-bit MSI is new! Anytime you install the Azure CLI, previously installed versions are updated automatically. This behavior allows you to try out the 64-bit install but reinstall the 32-bit MSI if you choose.

ZIP file Windows install

Beginning in [Azure CLI 2.57.0](#), the Azure CLI can be installed using a ZIP file in Windows environments. See the ZIP tab in [Install Azure CLI on Windows](#) for more information.

Tab completion in PowerShell

If you run the Azure CLI in PowerShell, tab completion is now available! Follow the instructions in [enable tab completion on PowerShell](#). The parameter values needed for PowerShell's `Register-ArgumentCompleter` command are provided in the article.

Tab completion is also available in [Azure Cloud Shell](#) and in most Linux distributions.

Sign in with Web Account Manager (WAM)

The Azure CLI now offers preview support for sign in with Web Account Manager (WAM). Read about the benefits of WAM and how to enable the feature in [Sign in with Web Account Manager](#).

Reference type and status

Reference type and status information is now available in Azure CLI reference content. Why is this important? Reference command status determines the support level.

You see this information in three places:

- New "type" and "status" columns in reference list tables.

Name	Description	Type	Status
Command name 1	Command description 1	Core	GA
Command name 2	Command description 2	Extension	Preview

For a live example, see the [reference index](#) or drill down to [az account](#).

- New status indicators under command names.



If there's no status indicator, the command group or reference command is GA (Generally Available). For a live example, see [az account subscription](#).

- New status indicator for parameters. Only deprecated parameters show a status. All other parameters inherit the status of the reference command.

For more information on Azure CLI statuses, see [Azure CLI terminology and support levels](#).

New onboarding tools

Are you new to the Azure CLI? Check out the new Onboarding Cheat Sheet to jump-start your journey, and find code examples in the A to Z indexes.

- [Onboarding cheat sheet](#)
- [Azure CLI conceptual article index](#)
- [Azure CLI sample index](#)

Azure CLI release notes

Article • 10/01/2024

October 01, 2024

Version 2.65.0

ACR

- Fix #21900: `az acr task identity remove`: Fix incorrect operation_group setting
- `az acr credential-set`: Improve help message

AD

- `az ad app create/update`: Add `--service-management-reference` argument

AKS

- `az aks nodepool delete-machines`: Add support to delete specific machines in an agent pool

App Config

- `az appconfig`: Add premium SKU support

App Service

- `az functionapp config set`: Map parameters explicitly for the internal `update_site_configs` function call

Compute

- `az vm/vmss create`: Add `--ssh-key-type` parameter to support generating Ed25519 SSH keys
- `az vm image list`: Expose `ImageDeprecationStatus` property in output

Container app

- Fix #29849: `az containerapp job stop`: Fix stop execution response
- Fix #29166: `az containerapp compose create`: Support image with tag and support get ACR from image

Cosmos DB

- `az cosmosdb restore`: Add `--disable-ttl` to support for Restore with Time-To-Live Disabled

DataLake

- `az dla`: Deprecate datalake analytics

HDInsight

- `az hdinsight update`: Add parameter `--assign-identity-type` to allow to update to `SystemAssigned` identity.

Monitor

- `az monitor action-group create`: Adjust formatting of `--actions` help message
- `az monitor action-group update`: Adjust formatting of `--add-actions` help message
- `az monitor action-group notification create`: Adjust formatting of `--add-actions` help message

MySQL

- `az mysql flexible-server create/update`: Add `--accelerated-logs` to support configuring accelerated logs for Business Critical tier

NetAppFiles

- `az netappfiles volume create`: Fix subnet id as resource id

Network

- `az network vnet peering`: Refine command interface of subnet peering

Packaging

- Release Docker image based on Azure Linux 3.0
- Include `jq` in Azure Linux docker image
- Drop Debian 10 support
- Use Azure Linux 3.0 (`mcr.microsoft.com/azurelinux/base/core:3.0`) as the base image for docker image tag `latest` and `2.65.0`

Profile

- `az login`: Fall back to device code flow in GitHub Codespaces

RDBMS

- `az postgres flexible-server update`: Support case-insensitive input for `--tier`, `--performance-tier`, `--sku`, and `--maintenance-window`
- `az postgres flexible-server migration create`: Add AWS_AURORA as a migration source type for PostgreSQL

SQL

- `az sql elastic-pool create`: Set `min_capacity` to `None` for non-serverless SKUs
- `az sql mi create/update`: Add `gpv2` parameter and `iops` parameter

September 03, 2024

Version 2.64.0

ACR

- `az acr helm`: Improve deprecation message

AKS

- `az aks nodepool update`: Add `--enable/disable-fips-image` flags for GA mutable fips
- `az aks create/update`: Support UserAssigned Managed Identity for grafana linking in managed prometheus

- `az aks create/update`: Update the VM SKU regex validation to include larger set of VMs for Azure Container Storage

App Service

- `az staticwebapp create/update`: Add `Dedicated` as supported SKU
- `az webapp config set`: Add MinTlsCipherSuite support
- `az webapp config access-restriction add/remove`: Allow skipping service tag validation

ARM

- `az ts create`: Fix incorrect handling of whitespace in string values

ARO

- `az aro create`: Add feature for adding Multiple Public IPs on cluster load balancer
- `az aro update`: Add feature for adding/removing Multiple Public IPs on cluster load balancer

Compute

- `az vm update`: Add support of Gen1 VM to trusted launch upgrade
- `az capacity reservation`: GA command group

Container app

- Fix #28998: `az containerapp env workload-profile add/update`: Fix `NoneType` object is not iterable error when environment doesn't enable workload profile
- `az containerapp create/up`: Make `--target-port` optional
- `az containerapp env create`: Deprecate unused argument `--docker-bridge-cidr`
- `az containerapp job stop`: Deprecate option to stop list of given job executions
- Fix #29711: `az containerapp logs show`: Fix JSON escaping by default or with `--format json`

HDInsight

- `az hdinsight create`: Support setting IP tags when creating HDInsight cluster
- `az hdinsight update`: Support updating the managed identity of the cluster

- `az hdinsight azure-monitor-agent show`: Support getting the managed identity of the cluster
- `az hdinsight azure-monitor-agent enable`: Enable the Azure Monitor Agent logs integration on an HDInsight cluster
- `az hdinsight azure-monitor-agent disable`: Disable the Azure Monitor Agent logs integration on an HDInsight cluster

MySQL

- `az mysql flexible-server maintenance`: New command group for managing maintenance of MySQL flexible server
- `az mysql flexible-server update`: Remove storage passing for server update

NetAppFiles

- `az netappfiles volume create/update`: Update max for `--usage-threshold`

Network

- Fix #29565: `az network nat gateway create`: Add `--tags` parameter

Packaging

- Support Python 3.12
- Release Azure Linux 3 RPM package
- [BREAKING CHANGE] Use CBL-Mariner 2.0 (mcr.microsoft.com/cbl-mariner/base/core:2.0) as the base image for docker image tag `latest` and `2.64.0`

Redis

- `az redis create/update`: Add `--disable-access-keys` to support disabling auth through access keys

SQL

- `az sql mi create/update`: Add `--authentication-metadata` to support auth metadata configuration

Storage

- `az storage account create/update`: Support account cold-tier
- `az storage copy/remove`: Add `--auth-mode login` support
- `az storage share delete`: Add `include-leased` for `--delete-snapshots`

August 06, 2024

Version 2.63.0

ACR

- `az acr login`: Support optional resource group argument to skip subscription resource scan

AKS

- `az aks create`: Not set the `--network-plugin` based on the default from the Python SDK
- `az aks create/update`: Support in place param updates for managed Prometheus
- `az aks create/update`: Add validations for PremiumV2 disk
- `az aks upgrade`: Support tier switch with AKS upgrade
- `az aks create/update`: Add `--network-policy none` option to command
- `az aks create`: Remove unsupported scenario command in help message

App Config

- `az appconfig import/export/restore`: Add correlation request id to bulk operations

App Service

- `az webapp list-runtimes`: Add parameter `--show-runtime-details` to show detailed runtime stacks and update the format of java related stacks listed
- `az webapp create`: Add parameter `--acr-identity` to allow users to choose user assigned identity for ACR image pull
- `az webapp config set`: Add parameter `--acr-use-identity` and `--acr-identity` to allow users update ACR image pull related configs

ARM

- `az stack group/sub/mg create`: Minor improvements to the confirmation message when updating an existing stack

CDN

- Fix #28717: `az afd secret`: Change the way to access parameter
- `az cdn portal-migration`: Add command group for classic CDN profile migration

Compute

- `az sig create/update`: Hide `--soft-delete` parameter in help messages
- Fix #29006: `az ssh`: Fix the `Permissions 0644 for '...' are too open` error
- `az vmss update`: Add new parameter `--enable-auto-os-upgrade` to support updating automatic OS upgrade policy argument
- `az vmss update`: Add new parameter `--upgrade-policy-mode` to support updating upgrade policy mode

Container app

- Fix #26688: `az containerapp up`: Fix logic about updating an existing containerapp
- `az containerapp job stop`: Return custom message for stop job execution
- Fix #29330: `az containerapp auth update`: Fix split logic for `--set`

Core

- Resolve CVE-2024-39689
- Resolve CVE-2024-6345
- Support `az config set extension.dynamic_install_allow_preview` for extension dynamic installation configuration
- `az logout`: Remove service principal access tokens from token cache

Cosmos DB

- `az cosmosdb delete`: Support `--no-wait`

MySQL

- `az mysql flexible-server import stop-replication`: Stop replication between source single server and target flexible server

NetAppFiles

- `az netappfiles account create`: Change `--key-vault-resource-id` to be optional

Network

- `az network custom-ip-prefix create`: Add parameter `--is-parent`
- `az network network-watcher connection-monitor`: Support to create connection monitor v2
- `az network vnet peering`: Support virtual network subnet peering

Packaging

- Drop Alpine Docker image

RDBMS

- [BREAKING CHANGE] `az postgres flexible-server create`: Update default value of PG version to be 16
- Fix #27422: `az postgres flexible-server create`: Bug fix for using existing subnet while creating pg flex server
- `az postgres flexible-server restore`: Bug fix when using resource id as value for source-server argument

Role

- `az role assignment list`: Add warning for classic administrators retirement

Service Connector

- `az containerapp connection create containerapp`: Support ACA2ACA connection

SQL

- `az sql midb move/copy`: Add destination subscription Id for managed database move/copy

- `az sql mi create`: Add `--dns-zone-partner` optional parameter

Storage

- `az storage fs directory upload/download`: Add back `--auth-mode login` as AzCopy supports Oauth now
- `az storage blob sync`: Add back `--auth-mode login` as AzCopy supports Oauth now

Synapse

- `az synapse spark job submit`: Add optional `--python-files` argument to support job submission

July 09, 2024

Version 2.62.0

ACR

- `az acr build/task`: Highlight the difference between arguments in the description

AKS

- `az aks create/update`: Update AMW resource.location to only accept alphanumeric characters
- `az aks create/update`: Add Azure Managed Grafana support for managed prometheus addon in usnat
- `az aks create/update`: Ephemeral disk additional support for Azure Container Storage
- `az aks create/update`: Add and fix existing validations for ephemeral disk support for Azure Container Storage

App Config

- `az appconfig`: Update help message for App Configuration store
- `az appconfig kv import`: Update the help message for ignore-match

App Service

- `az webapp deploy`: Disable deploymentstatus API for deployments to app slots
- `az webapp deployment source config-zip`: Disable deploymentstatus API for deployments to app slots
- Fix #29041: `az webapp config access-restriction add`: Fix edge case validating service tags
- `az functionapp create`: Avoid throwing exception when endOfLifeDate not passed from stacks API

ARM

- `az stack mg create`: Validate MG stacks automatically before creation
- `az bicep restore`: Fix typos in help messages

Backup

- `az backup restore restore-disks/restore-azurewl/restore-azurefiles/restore-azurefileshare`: Add `--tenant-id` for cross-tenant MUA protection, otherwise allows restore when protected by a resource guard
- `az backup vault encryption update`: Add support for MUA for CMK operations

CDN

- Fix #28721: `az afd origin-group create`: Add parameter to support session affinity state
- Fix #28824: `az cdn profile`: Add default location
- Fix #28733: `az cdn endpoint update`: Add support to update an existing endpoint

Compute

- `az vmss update`: Add new parameter `--zones` to support setting vmss zonal expansion
- `az vm install-patches`: Add new parameter `--max-patch-publish-date` to support setting max patch publish date
- `az vmss reimagine`: Add new parameter `--force-update-os-disk-for-ephemeral` to support updating the base OS disk
- `az vmss create/update`: Add new parameter `--scheduled-event-additional-publishing-target-event-grid-and-resource-graph` to support setting scheduled event policy

- `az vmss create/update`: Add new parameter `--enable-user-reboot-scheduled-events` to support setting policy for rebooting scheduled event
- `az vmss create/update`: Add new parameter `--enable-user-redeploy-scheduled-events` to support setting policy for redeploying scheduled event
- `az vmss update`: Add an option `Standard` to `--security-type` parameter
- `az vmss create`: Support set upgrade policy mode to automatic during flexible VMSS creation
- `az sig image-definition create`: Fix default value of security type

Container app

- `az containerapp env create/update`: Support peer-to-peer traffic encryption with `-enable-peer-to-peer-encryption`
- `az containerapp job update`: Fix poll logic and `--no-wait`
- `az containerapp show-custom-domain-verification-id`: Show verification id used for binding custom domain
- `az containerapp env update`: Fix logs configuration about removing destination with `--logs-destination none`
- `az containerapp list-usages`: Support list-usages in subscription
- Fix #28983: `az containerapp job registry`: Support commands for modifying the job's registries
- `az containerapp auth`: Support Token Store with `--token-store`, `--sas-url-secret`, `--sas-url-secret-name`, `--yes`
- Fix #29128: `az containerapp env certificate`: Support managed certificate
- Fix #29172: `az containerapp secret set`: Remove the limit of length for secret name

Core

- Fix #28997: Fix error "User cancelled the Accounts Control Operation" when logging in with WAM as Administrator
- Resolve CVE-2024-37891
- Resolve CVE-2024-35195
- `aaz`: Add argument completer support

Cosmos DB

- `az service create`: Add `--gateway-type` to support multiple dedicated gateway types

MySQL

- `az mysql flexible-server update`: Support replica resource group different with primary server
- `az mysql flexible-server detach-vnet`: New command to support converting a vnet azure mysql server to a non-vnet server
- `az mysql flexible-server deploy/update`: Add warning message for HA enabled and input validator

Network

- `az network application-gateway waf-policy custom-rule update`: Add parameter `--js-cookie-exp-time`
- `az network cross-region-lb address-pool`: Add parameter `--admin-state`
- `az network application-gateway rewrite-rule`: Add `--request-header-configurations` and `--response-header-configurations`
- `az network virtual-appliance restart`: Allow restart of virtual machines associated with a network virtual appliance
- `az network virtual-appliance inbound-security-rule`: Support of GET operation for NVA Inbound Security Rule
- `az network vnet subnet`: Add parameter `--endpoints`

RDBMS

- `az postgres flexible-server restore`: Add capability to restore to new server using Premium SSD v2 Disks by setting Storage Type to "PremiumV2_LRS"
- `az postgres flexible-server migration create`: Fix bugs while passing in MigrationRuntimeResourceId to Migration Parameters, now need to be provided within properties json file
- `az postgres flexible-server firewall-rule create`: Correct firewall rule name and ip range validators
- `az postgres flexible-server update`: Add argument to enable or disable public-access
- `az postgres flexible-server create`: Add argument `--create-default-database` to support disabling default database creation
- `az postgres flexible-server upgrade`: Unblock MVU for Burstable from CLI
- `az postgres flexible-server update`: Correct setting `--maintenance-window` to be disabled

SignalR

- `az signalr upstream update`: Remove the empty location string when updating resource

SQL

- `az sql db update`: Add `--manual-cutover` and `--perform-cutover` arguments

Storage

- `az storage container policy`: Fix cases where permission and expiry are both null
- `az storage account create/update`: Support TLS 1.3, update recommended `--min-tls-version` to be 1.2

May 21, 2024

Version 2.61.0

AKS

- [BREAKING CHANGE] `az aks create`: Specifying `--enable-managed-identity` and `--service-principal`/`--client-secret` at the same time will cause a `MutuallyExclusiveArgumentError`
- [BREAKING CHANGE] `az aks create`: Change the default value of option `--enable-managed-identity` from `True` to `False`
- `az aks mesh upgrade rollback/complete`: Add `--yes` parameter to support not prompting the users to confirm the operation
- `az aks create/update`: Add `SecurityPatch` option to `--node-os-upgrade-channel` parameter
- `az aks create/update`: Add new parameter `--enable-cost-analysis` to enable exporting Kubernetes namespace and deployment details to the Cost Analysis views
- `az aks create`: Backfill the value of `--enable-managed-identity` to `True` when options `--service-principal` and `--client-secret` are not specified at the same time
- `az aks nodepool update`: Add option `--os-sku` to support updating os sku in place
- `az aks create`: Add `--ampls-resource-id` and `--enable-high-log-scale-mode` optional parameters for Monitoring Addon

- `az aks enable-addons`: Add `--ampls-resource-id` and `--enable-high-log-scale-mode` optional parameters

App Service

- [BREAKING CHANGE] `az webapp deploy`: Use deployment status API for deployment output for Linux Web Apps
- [BREAKING CHANGE] `az webapp up`: Use deployment status API for deployment output for Linux Web Apps
- [BREAKING CHANGE] `az webapp deployment source config-zip`: Use deployment status API for deployment output for Linux Web Apps
- `az functionapp scale config always-ready`: Set alwaysReady property to empty array if it is null
- `az functionapp`: Update messaging for flex function apps
- `az functionapp deployment source config-zip`: Allow users with no Microsoft.Web/serverFarm read privileges to deploy function apps
- `az webapp list`: Fix the bug `--show-details` fails while resource group name is not specified
- `az webapp list-runtimes/create/up`: Add Java 21 support
- `az functionapp create`: Use stacks API netFrameworkVersion value instead of the default value from the Python SDK
- `az functionapp create`: The linuxFxVersion for dotnet-isolated linux consumption apps will no longer be left empty
- `az functionapp`: Not block execution of command when runtime cannot be detected, and omit showing warning for runtime when not applicable (e.g. centauri apps, apps running a docker image)
- `az appservice plan create/update`: Add IsolatedV2 memory intensive SKU support
- `az functionapp create`: If customers do not provide an image when creating a Centauri function app, we use the updated default Centauri image

ARM

- [BREAKING CHANGE] `az stack group/sub/mg create/delete`: Remove the deprecated `--delete-all`, `--delete-resources`, and `--delete-resource-groups` flags. Use the `--action-on-unmanage` / `--aou` parameter instead
- `az group delete`: Add new option `Microsoft.Databricks/workspaces` for `--force-deletion-types` parameter
- `az deployment`: Support inline parameters with `.bicepparam` in single `--parameters` argument

- `az stack group/sub/mg validate`: Add new `validate` command to perform preflight validation on a stack deployment
- `az stack group/sub create`: Validation of a stack will now occur before a stack is created or updated
- `az stack group/sub/mg create/delete`: Action on unmanage behavior for stack managed management groups can now be configured
- `az stack group/sub/mg create`: The correlation ID of the create operation is now returned as a property of the stack
- `az stack group/sub/mg create/delete`: Add new flag `--bypass-stack-out-of-sync-error` / `--bse` that will bypass errors related to the resource list of a stack being out of sync

Compute

- [BREAKING CHANGE] `az sig image-definition create`: Set the default values for Hyper-V generation and Security Type
- `az vmss create/update`: Add new parameters `--enable-resilient-creation` and `--enable-resilient-deletion` to support Resiliency Policy on VMSS
- `az vm create/update`: Add new option `NvmeDisk` for `--ephemeral-os-disk-placement` parameter
- `az vmss create/update`: Add new option `NvmeDisk` for `--ephemeral-os-disk-placement` parameter
- `az vm create`: Add new parameters `--source-snapshots-or-disks` and `--source-snapshots-or-disks-size-gb` to support implicit disk creation from snapshot and disk
- `az vm create`: Add new parameters `--source-disk-restore-point` and `--source-disk-restore-point-size-gb` to support implicit disk creation from disk restore point
- `az vmss update`: Add new parameter `--ephemeral-os-disk` to support in-place mutual migration of VMSS from ephemeral to non-ephemeral OS disk
- `az vmss update`: Add new parameter `--ephemeral-option` to support setting ephemeral disk setting

Compute Diagnostic

- `az compute-recommender spot-placement-recommender`: Add new command to support generating placement scores for Spot VM SKU

Container app

- `az containerapp create/update`: Fix `--scale-rule-tcp-concurrency` for TCP scale rule
- `az containerapp compose create`: Fix issue where the environment's location is not resolved from `--location`
- Fix #28864: `az containerapp ingress update`: Fix updating transport from http to tcp with `--transport tcp`
- `az containerapp compose create`: Fix variable mixing issue when `--compose-file-path` contains multiple services
- Fix #28380: `az containerapp ingress access-restriction set`: Fix `KeyError` when `name` not exists

Core

- Resolve CVE-2024-34064
- [BREAKING CHANGE] `az login`: Use WAM as the default authentication method on Windows. For more details, see <https://go.microsoft.com/fwlink/?linkid=2271136>
- `aaz`: Support extended values in `AAZArgEnum`
- Add `tenantDefaultDomain` and `tenantDisplayName` properties to login contexts (shown by `az account list`)
- Prompt for interactive authentication for all silent authentication failures

Key Vault

- [BREAKING CHANGE] `az keyvault create`: Default `--enable-rbac-authorization` to true
- `az keyvault key create`: Update the release policy used for `--default-cvm-policy`

NetAppFiles

- [BREAKING CHANGE] `az account backup`: Replace backup commands with backup-vault commands
- [BREAKING CHANGE] `az volume backup status`: Remove `volume backup status` command, replace with `az netappfiles volume latest-backup-status show`
- `az netappfiles account backup-vault`: Add backup vault command group
- `az netappfiles volume latest-backup-status show`: Add command to get latest backup status

- `az netappfiles volume latest-restore-status show`: Add command to get latest backup status
- `az netappfiles resource region-info list`: Add command to list region specific information
- `az netappfiles resource region-info default show`: Add command to get storage to network proximity and logical zone mapping information

Network

- [BREAKING CHANGE] `az network dns zone`: Deprecate `--zone-type`, `registration-vnets` and `resolution-vnets`
- `az network vnet subnet`: Add parameter `--sharing-scope`
- `az network private-endpoint-connection`: Add `Microsoft.App/managedEnvironment` for private endpoint connections
- Fix #28615: `az network application-gateway address-pool update`: Race condition in concurrent scenario
- Fix #28705: `az network lb rule`: Authentication token not being generated

Packaging

- Add Ubuntu 24.04 Noble Numbat support
- [BREAKING CHANGE] Drop Ubuntu 18.04 support

Profile

- `az login`: Introduce login experience v2. For more details, see <https://go.microsoft.com/fwlink/?linkid=2271236>

RDBMS

- `az postgres flexible-server migration create`: Add private endpoint support for migrations by providing migration runtime resource ID as command line argument

Security

- [BREAKING CHANGE] `az security contact create`: Deprecate `--alerts-admins` and `--email`
- [BREAKING CHANGE] `az security setting update`: Deprecate `--enabled`

Service Connector

- `az aks connection list/show`: Add kubernetes resource name
- `az source connection create cognitiveservices`: Support OpenAI/AIServices/CognitiveServices as target
- `az webapp connection list`: Fix interactive input

Storage

- [BREAKING CHANGE] `az storage account update`: Prompt user for possible charge increases when changing `--access-tier`, add `--upgrade-to-storagev2`
- [BREAKING CHANGE] `az storage container set-permission/get-permission` and `az storage container policy`: Remove `--auth-mode login` and `--sas-token` for container access policy commands as only shared key authorization is supported on server side

April 30, 2024

Version 2.60.0

ACR

- [BREAKING CHANGE] `az acr connected-registry create`: Mode default value change from ReadWrite to ReadOnly
- `az acr connected-registry create`: If data-endpoint disabled ask for confirmation to enable it instead of throwing an error

AKS

- `az aks create/update`: Prompt warning during disablement about CR deletion
- `az aks create/update`: Update RP registration code to work on azure monitor subscription
- `az aks create/update`: Update to add default region for workspace creation in air gapped cloud
- `az aks nodepool add`: Add parameter `--disable-windows-outbound-nat` to add a Windows agent pool which the Windows OutboundNAT is disabled

App Service

- `az webapp config container set`: Remove docker prefix and rename container related parameters
- `az webapp create` and `az webapp deployment slot create`: Rename `--docker-registry-server-user` and `--docker-registry-server-password` to `--container-registry-user` and `--container-registry-password`
- `az webapp create` and `az webapp deployment slot create`: Add parameters `--container-image-name` and `container-registry-url` to deprecate `--deployment-container-image-name`
- `az webapp create` and `az logicapp create`: Update help message
- `az webapp config set`: Add new parameter `--runtime` to allow users to update their stack by using single parameter
- `az functionapp create`: Remove unnecessary app settings for flex function apps
- Fix #28588: `az webapp config access-restriction add`: Check for null before getting values
- `az webapp config access-restriction set`: Add new parameter `--default-action` to configure default action for main site
- `az webapp config access-restriction set`: Add new parameter `--scm-default-action` to configure default action for scm site
- `az webapp list`: Add parameter `--show-details` to include detailed site configuration of listed webapps in output
- `az functionapp create`: Will validate that the provided runtime is supported by flex when creating a flex function app
- `az functionapp list-flexconsumption-runtimes`: Add support for this new command so that it provides the list of supported flex runtimes when provided the `--location` and `--runtime`

ARM

- `az bicep install`: Support additional process architectures with Bicep CLI
- `az deployment`: Return better message on incorrect bicepparam file path
- `az bicep format`: Replace `--newline` with `--newline-kind`
- `az bicep publish`: Replace `--documentationUri` with `--documentation-uri`

Backup

- `az backup vault update`: Take `--tenant-id` as input for resolving cross-tenant resource guard scenarios. Vault Immutability feature also has resource guard protection now, and support for the same has been added.

- `az backup protection disable`: Take `--tenant-id` as input for resolving cross-tenant resource guard scenarios. Stop protection feature also has resource guard protection now, and support for the same has been added.

Batch

- `az batch pool create`: Add `--upgrade-policy-mode` argument to support automatic OS upgrade
- `az batch pool create`: Add `--enable-auto-os-upgrade` argument to enable automatic OS upgrade
- `az batch pool create`: Add `--disable-auto-rollback` argument to disable OS image rollback feature
- `az batch pool create`: Add `--defer-os-rolling-upgrade` argument to defer OS upgrades on the TVMs if they are running tasks
- `az batch pool create`: Add `--use-rolling-upgrade-policy` argument to support OS rolling upgrade policy
- `az batch pool create`: Add `--enable-cross-zone-upgrade` argument to support cross zone OS upgrade
- `az batch pool create`: Add `--max-batch-instance-percent` argument to set the maximum percent of total VMs that will be upgraded in one batch
- `az batch pool create`: Add `--max-unhealthy-instance-percent` argument to set the maximum percentage of the total VMs can be simultaneously unhealthy
- `az batch pool create`: Add `--max-unhealthy-upgraded-instance-percent` argument to set the maximum percentage of upgraded VMs that can be found to be in an unhealthy state
- `az batch pool create`: Add `--pause-time-between-batches` argument to set the wait time between batches in rolling OS upgrade
- `az batch pool create`: Add `--prioritize-unhealthy-instances` argument to support upgrade all unhealthy VMs first
- `az batch pool create`: Add `--rollback-failed-instances-on-policy-breach` argument to enable rollback failed instances to previous model if the Rolling Upgrade policy is violated

Compute

- `az vmss create`: Add support of configure the rolling mode upgrade policy during VMSS creation
- `az vmss update`: Add new parameter `--max-surge` to support updating rolling upgrade policy max surge

- `az capacity reservation group list`: Add new parameter `--resource-ids-only` to support retrieving the capacity reservation group resource ids
- `az capacity reservation group create`: Change `--sharing-profile` to support unsharing subscriptions by passing nothing

Containerapp

- `az containerapp env create`: Support `--dapr-connection-string` to set application insights connection string used by Dapr to export service to service communication telemetry
- Fix #28553: `az containerapp exec`: Fix the error of `inappropriate ioctl for device`

Core

- Fix #28737: Fix token cache for service principal authentication
- Resolve CVE-2023-3651

Monitor

- `az monitor log-analytics workspace update`: Add parameter `--sku-name`

NetAppFiles

- `az netappfiles volume-group create`: Add `--zones` argument to set Availability Zone for volume group volumes
- `az netappfiles volume create/update`: Update maximum value for `--usage-threshold` to support large volumes

Network

- `az network virtual-appliance inbound-security-rule`: Support for Permanent Inbound Security Rule

RDBMS

- `az postgres flexible-server upgrade`: Add capability to perform major version upgrade to PG16

Service Connector

- `az aks connection`: AKS support for Service Connector
- `az webapp/containerapp/spring connection create/update`: Store configurations in App Config
- `az source connection create`: Support auth opt out

Service Fabric

- `az sf managed-cluster network-security-rule`: Add new network security rule commands

SQL

- `az sql server create/update`: Add 1.3 to minTLSVersion and make TLS 1.2 as default

April 02, 2024

Version 2.59.0

ACR

- Fix #14768: `az acr login`: Add environment variable for docker command

ACS

- `az aks create`: Add flag `--enable-app-routing` to enable app routing
- `az aks approuting`: Add command group to handle enable/disable/update of the app routing addon
- `az aks approuting zone`: Add command group to handle add/delete/update/list actions of DNS zone resources associated to the approuting addon
- `az aks create/update`: Introduce changes for Azure container storage in ACS CLI

AD

- `az ad`: Rename Azure Active Directory to Microsoft Entra ID

AKS

- `az aks create`: Add optional parameter `--revision` to set revision for the Azure Service Mesh addon while creating AKS cluster

- `az aks mesh get-upgrades`: Fix command failure with a traceback if ASM addon is disabled
- `az aks create/update`: Enable mooncake support for managed prometheus addon
- `az aks create/update`: Block Azure Managed Grafana for managed prometheus addon in air gapped cloud
- `az aks create`: Correct use of "comma-separated" in help

App Config

- `az appconfig feature filter update`: GA command
- `az appconfig kv export`: GA parameter `--export-as-reference`

App Service

- `az functionapp create`: Add support for Node 20 for Flex function apps
- `az functionapp create`: Make Node 20 the default for node flex function apps and Python 3.11 the default for python flex function apps
- `az functionapp create`: Add support for SystemAssignedIdentity and UserAssignedIdentity as the deployment storage authentication type
- `az webapp update`: Add new parameter `--elastic-web-app-scale-limit` and scaling parameter options
- `az appservice plan update`: Add new parameter `--elastic-web-app-scale-limit` and scaling parameter options
- `az webapp deployment source config-zip`: Mark this command as deprecated, recommend using the `az webapp deploy` command instead of it

ARM

- `az stack group create`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage` argument
- `az stack group delete`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage` argument
- `az stack sub create`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage` argument
- `az stack sub delete`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage`

argument

- `az stack mg create`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage` argument
- `az stack mg delete`: Deprecate the `--delete-resources`, `--delete-resource-groups` and `--delete-all` options and redirect to the new `--action-on-unmanage` argument
- `az deployment`: Treat nullable parameters as non-required for Bicep deployment

ARO

- `az aro create/validate`: Fix bug in permissions validation that was preventing cluster creation in cases where the invoking user had the necessary permissions

CDN

- `az afd profile`: Add parameter `--identity`

Compute

- `az snapshot grant-access`: Add parameter `--file-format` to support specifying file format when making request for SAS on a VHDX file format snapshot
- `az vmss create`: Add `--enable-auto-os-upgrade` parameter to support automatic OS Upgrade while creating VMSS
- `az sig image-definition create`: Add warning message for Hyper-V generation and Security Type
- `az vmss create/update`: Add parameters to specify the security posture to be used for all virtual machines in the scale set
- `az capacity reservation group create/update`: Add new parameter `--sharing-profile` to support sharing capacity reservation group across subscriptions
- `az snapshot create`: Add parameter `--bandwidth-copy-speed` to allow a snapshot to be copied at a quicker speed

DataBoxEdge

- `az databoxedge device`: Add command group `share` to support managing device share
- `az databoxedge device`: Add command group `user` to support managing device user
- `az databoxedge device`: Add command group `storage-account` to support managing device storage account

- `az databoxedge device`: Add command group `storage-account-credential` to support managing device storage account credential
- `az databoxedge device`: Add command `get-extended-information` to support getting extended information

MySQL

- `az mysql flexible-server advanced-threat-protection-setting show`: Show server's advanced threat protection setting
- `az mysql flexible-server advanced-threat-protection-setting update`: Update server's advanced threat protection setting using `--state` as Enabled/Disabled
- `az mysql flexible-server import create`: Add support for online migration for single to flex

NetAppFiles

- `az netappfiles check-file-path-availability`: Add new command to check if a file path is available
- `az netappfiles check-name-availability`: Add new command to check if a resource name is available
- `az netappfiles check-quota-availability`: Add new command to check if a quota is available
- `az netappfiles query-network-sibling-set`: Add new command to describe a network sibling set
- `az netappfiles update-network-sibling-set`: Add new command to update the network features of a network sibling set
- `az netappfiles quota-limit`: Add new command group to manage quota limits
- `az netappfiles volume populate-availability-zone`: Add new command to populate availability zone information for a volume
- `az netappfiles volume replication re-initialize`: Add new command to re-establish a previously deleted replication between 2 volumes that have a common ad-hoc or policy-based snapshots

Network

- `az network virtual-appliance connection`: Add update command for NVA connection
- `az network dns record-set`: Add `--traffic-management-profile` for TMLink recordset feature

- `az network application-gateway waf-policy`: Change default rule set from CRS3.0 to DRS2.1
- `az network virtual-appliance`: Add `--internet-ingress-ips` and `--network-profile`

March 05, 2024

Version 2.58.0

ACR

- `az acr`: Update registry and webhook example names to be lowercase
- `az acr create/update`: Add new argument `--allow-metadata-search` to enable the metadata-search feature for the registry

AKS

- Add monitoring addon to support default workspace in AGC
- `az aks get-versions`: Show extra column on supportPlan
- `az aks create/update`: Update region map for default region creation with new Azure Monitor Workspace regions
- `az aks update`: Zero can be set to outbound-ports,outbound-ip-count in loadbalancer profile and outbound-ip-count in natgateway profile in AKS

API Management

- `az apim api export`: Add command to export an API Management API

App Service

- `az webapp up/create/update`: Add new parameter `--basic-auth` to allow users to enable and disable basic auth

ARM

- Fix #27855: `az bicep generate-params`: Bicep install messages sent to stdout

Backup

- [BREAKING CHANGE] `az backup item set-policy`: Add warning prompt for migration from Standard to Enhanced Policy

Batch

- `az batch pool create`: Add new parameter `--resource-tags` to support specifying resource tags for the pool. Any resource created for the pool will then also be tagged by the same resource tags
- `az batch pool create`: Add new parameters `--security-type`, `--encryption-at-host`, `--secure-boot-enabled`, and `--v-tpm-enabled` to support Trusted Launch Security Type for VMs/VMSS deployments
- `az batch pool create`: Add new parameters `--caching`, `--disk-size-gb`, `--write-accelerator-enabled`, and `--storage-account-type` to support Batch Node Agent temp disk-less SKUs

CDN

- Fix #28240: `az afd rule create`: Cannot create without condition
- Fix #28223: `az afd route create`: Cannot create without `--content-types-to-compress`
- Fix #27744: `az afd origin-group`: Add parameter `--enable-health-probe`

Compute

- `az vmss nic`: Update help messages to guild users to specific commands for Flexible VMSS
- `az vm host redeploy`: Add command to redeploy the dedicated host
- Fix #28397: `az vm create`: Fix creating VM with `--security-type Standard`
- `az vmss application set`: Fix updates to the purchase plan are not supported when updating VMSS
- `az vmss update-domain-walk`: Add new command to support updating vm in a service fabric vmss

Containerapp

- `az containerapp revision copy`: Fix `--from-revision` bug for inheriting a specific revision contains scale rules
- `az containerapp update`: Fix TypeError: Argument of type 'NoneType' is not iterable

- Fix #28226: `az containerapp job update`: Update existing scale rules if `--scale-rule-name` is passed

Core

- [BREAKING CHANGE] `az login`: Replace `allow_broker` with `enable_broker_on_windows` (#27726)

Cosmos DB

- `az cosmosdb sql database/container restore`: Fix support for restore of deleted database resource in the same SQL account
- `az cosmosdb mongodb database/collection restore`: Fix support for restore of deleted database resource in the same MongoDB account
- `az cosmosdb gremlin database/graph restore`: Fix support for restore of deleted database resource in the same gremlin account
- `az cosmosdb table restore`: Fix support for restore of deleted table resource in the same account

MySQL

- `az mysql flexible-server import create`: Change the default progress message from starting to running

Network

- `az network virtual-appliance connection`: Add show and list commands for NVA connection
- `az network vnet-gateway`: Add parameters `--allow-remote-vnet-traffic` and `--allow-vwan-traffic`
- `az network express-route gateway`: Add parameter `--allow-non-vwan-traffic`

RDBMS

- Fix #27713: `az postgres flexible-server list-skus -o table`: Fix table output from list-skus command

Security

- `az security api-collection`: Manage Azure API Management API connections to Microsoft Defender for APIs
- `az security security-connector`: Manage cloud security posture management (CSPM) and cloud workload protection (CWP) across multicloud resources

SQL VM

- Fix #27300: `az sql vm group create`: `--cluster-subnet-type` should only be passed into `WsfcDomainProfile`

Storage

- Fix #28356: `az storage account or-policy update`: Fix `-p @policy.json` to allow passing in json files
- `az storage blob upload-batch`: Support `--tags`, make `--overwrite` non-preview

February 06, 2024

Version 2.57.0

AKS

- `az aks upgrade`: Add forceupgrade settings to aks stable cli
- `az aks mesh`: Add `az aks mesh` commands to manage Azure Service Mesh in given cluster
- `az aks nodepool add/update/upgrade`: Add new parameter `--node-soak-duration` to wait after draining a node and before reimaging it and moving on to next node

App Service

- `az functionapp create`: Use app insights connection string instead of instrumentation key
- `az webapp create`: Add new parameter `--acr-use-identity` to configure pull image from ACR using MSI when creating a container web app
- `az webapp up`: Add `--track-status` arg to use deploymentstatus API
- `az webapp deployment source config-zip`: Add `--track-status` arg to use deploymentstatus API
- `az functionapp create`: Only allow apps with functions version 4

Backup

- `az backup restore restore-disks`: Add support for disk restore in edge-zone backups
- `az backup restore restore-disks`: Support for Taiwan North and Taiwan North-West Cross region restore

Billing

- `az billing account`: Add new command `list-invoice-section` to support listing invoice section
- `az billing account`: Add new command group `invoice-section` to support managing account invoice section
- `az billing`: Add new command `enrollment-account` to support managing enrollment account
- `az billing invoice section`: Add new command `initiate-transfer` to support initiating transfer
- `az billing profile`: Add new command `reservation list` to support managing reservation
- `az billing`: Add new command group `transfer` to support managing transfer

Compute

- `az image builder error-handler`: Add group to manage error handling options upon a build failure
- `az sshkey create`: Add parameter `--encryption-type` to specify the encryption type of SSH keys to be generated
- `az vm monitor log show`: Deprecate `azure-loganalytics` and apply `azure-monitor-query`
- `az vm/vmss create`: Support Trusted Launch as default deployment option

Consumption

- Fix #20995 #23825: Update `az consumption` API version

Containerapp

- `az containerapp update`: Fix issue for minReplicas in `--yaml` or `--min-replicas` is not set when the value is 0

- `az containerapp up`: Fix issue when registry creds are provided for ACRs using `containerapp up` command

Key Vault

- `az keyvault create`: Add warning for upcoming breaking change that `--enable-rbac-authorization` will default to `true`
- `az keyvault secret set-attributes`: Fix 'Datetime with no tzinfo will be considered UTC' warning

Monitor

- `az monitor metrics`: Add `list-sub` and `list-sub-definition`
- `az monitor metrics list`: Fix `--top` help message

MySQL

- `az mysql flexible-server import create`: Add support for operation progress estimated completion time for import from physical backup from azure blob to flexible server

Network

- `az network vnet-gateway create/update`: Add parameter `--enable-private-ip`
- Fix #28131: `az network vnet-gateway list`: Conflict key when apply client flatten
- `az network express-route port authorization`: Support to manage authorization with express route port
- `az network private-link-service list-auto-approved`: Support to list all auto-approved private link services
- `az network public-ip ddos-protection-status show`: Support to get the DDoS protection status of public IP address
- `az network vnet-gateway`: Support ExpressRoute SKU
- `az network lb create`: Support cross-subscription resource ID

Packaging

- Support Windows ZIP package (Preview)

RDBMS

- `az postgres flexible-server private-endpoint-connection`: Add support for private endpoint connection commands like list, show, approve, reject, delete for PostgreSQL flexible server
- `az postgres flexible-server private-link-resource`: Add support for private link resource commands like list, show for PostgreSQL flexible server
- `az postgres flexible-server replica stop-replication`: Stop replication to a read replica and make it a read/write server. This command is deprecated. Use `az postgres flexible-server replica promote` instead.

Redis

- `az redis flush`: Add support for flush operation
- `az redis create`: Add support for Microsoft Entra Authentication
- `az redis access-policy/access-policy-assignment`: New command groups to manage access policy

Service Connector

- `az webapp/containerapp/spring connection create/update app-insights`: Support App Insights as target service
- `az webapp connection create sql`: Support auto install for serviceconnector-passwordless extension
- `az webapp/functionapp/containerapp/springapp connection`: Support Microsoft Entra ID auth types for table storage
- `az webapp/containerapp/spring connection create/update`: Support opt out action for configurationInfo and publicNetwork

SQL

- `az sql db ltr-policy/ltr-backup`: Add support for backup storage access tier
- `az sql db ltr-policy set`: Add the make-backups-immutable parameter to ltr policy
- `az mi create/update`: Add optional parameter `--database-format` and `--pricing-model` for SQL Managed Server resource
- `az sql mi refresh-external-governance-status`: Add a new command for Azure SQL Managed Instance refresh external governance status

Storage

- `az storage file copy start`: Fix when copying between two storage accounts and source file has no parent directory
- Fix #28202: `az storage container policy create`: Fix creating an access policy would delete public access permission
- `az storage container set-permission`: Fix failing when updating public access permission with an existing access policy
- Fix #21876: `az storage blob upload-batch/download-batch/delete-batch`: Add double quotes to `--pattern` value for unix shell

January 09, 2024

Version 2.56.0

AKS

- `az aks create/update`: Add new parameter `--load-balancer-backend-pool-type` to define the Load Balancer backend pool type of managed inbound backend pool
- `az aks create`: Add parameter `--node-public-ip-tags` to specify the ipTags of the node public IPs
- `az aks nodepool`: Add parameter `--node-public-ip-tags` to specify the ipTags of the node public IPs
- `az aks create` and `az aks nodepool add`: Add crg-id option to create nodepool with Capacity Reservation Group
- `az aks stop`: Add warning when private link cluster is stopped
- `az aks trustedaccess role`: Add new command group to manage trusted access roles
- `az aks trustedaccess rolebinding`: Add new command group to manage trusted access role bindings
- `az aks trustedaccess rolebinding list`: Add new command to list all the trusted access role bindings
- `az aks trustedaccess rolebinding show`: Add new command to get the specific trusted access role binding according to binding name
- `az aks trustedaccess rolebinding create`: Add new command to create a new trusted access role binding
- `az aks trustedaccess rolebinding update`: Add new command to update an existing trusted access role binding
- `az aks trustedaccess rolebinding delete`: Add new command to delete a trusted access role binding according to name

- `az aks update`: Add parameter `--network-plugin` to update the network plugin of the AKS cluster

App Config

- `az appconfig feature set`: Add requirement type to feature command

App Service

- `az functionapp create`: Add DAPR support for Centauri apps
- `az functionapp create`: Remove workarounds for Centauri
- `az webapp deploy`: Add `--track-status` arg to use deploymentstatus API
- `az functionapp create`: Add workload profile support for Centauri apps

ARM

- `az bicep publish`: Add optional parameter `--with-source` to publish source code with the module (experimental)
- `az lock delete`: Fix the case sensitive comparison issue for resource group name

Compute

- `az vmss create`: Change `--orchestration-mode` to support only `uniform` for old profile
- `az vm/vmss create/update`: Add new parameters `--enable-proxy-agent` and `--proxy-agent-mode` to support azure metadata security protocol
- `az vm/vmss create`: Add new option `NonPersistedTPM` for parameter `--os-disk-security-encryption-type` to support creating Intel TDX based Confidential VM

Containerapp

- `az containerapp job update`: Fix bug for minExecutions in `--yaml` is not set when the value is `0`
- `az containerapp hostname bind`: Fix an issue about parsing the environment's resource group when the `--environment` is a resource id

Cosmos DB

- `az cosmosdb sql database restore`: Add support for restore of deleted database resource in the same SQL account
- `az cosmosdb sql container restore`: Add support for restore of deleted container resource in the same SQL account
- `az cosmosdb mongodb database restore`: Add support for restore of deleted database resource in the same MongoDB account
- `az cosmosdb mongodb collection restore`: Add support for restore of deleted collection resource in the same MongoDB account
- `az cosmosdb gremlin database restore`: Add support for restore of deleted database resource in the same gremlin account
- `az cosmosdb gremlin graph restore`: Add support for restore of deleted graph resource in the same gremlin account
- `az cosmosdb table restore`: Add support for restore of deleted table resource in the same account

Eventhub

- `az eventhubs georecovery-alias create`: Name parameter is required

Extension

- `az extension`: Enable extension semantic versioning and join `experimental` into `preview`
- `az extension add/update`: Add `--allow-preview` to distinguish stable-only installation with preview-included installation

Key Vault

- `az keyvault create/update-hsm`: GA `--mi-user-assigned`
- `az keyvault backup/restore start`: GA `--use-managed-identity`

Monitor

- `az monitor log-analytics workspace`: Add `list-link-target` and `list-available-service-tier`

Network

- `az network application-gateway waf-config list-dynamic-rule-sets`: Support to get the WAF dynamic manifest
- `az network nsg rule list`: Fix ranges and prefixes disappeared with `-o table` option
- `az network vnet subnet`: Add `--private-endpoint-network-policies` and `--private-link-service-network-policies`

RDBMS

- `az postgres flexible-server virtual-endpoint`: Add support for virtual endpoints for PostgreSQL flexible server
- `az postgres flexible-server replica promote`: Add capability to stop replication and promote to primary or standalone server with the selection of planned/force data syncs.
- `az postgres flexible-server server-logs list`: List server log files for PostgreSQL flexible server
- `az postgres flexible-server server-logs download`: Download server log files for PostgreSQL flexible server
- `az postgres flexible-server create`: Add capability to set storage type to PremiumV2_LRS and provide values for IOPS and Throughput during creation
- `az postgres flexible-server update`: Add capability to update the values of IOPS and Throughput during update
- `az postgres flexible-server migration create`: Add migration option like Migrate, Validate and ValidateAndMigrate using parameter `--migration-option` and json file for Migration configuration to support additional properties like sourceType and sslMode

Redis

- `az redis import/export`: Add `--storage-subscription-id` support while importing/exporting data

Service Bus

- `az servicebus georecovery-alias create`: Name parameter is required

Storage

- [BREAKING CHANGE] `az storage account create`: Server change default value for `--allow-blob-public-access` and `--allow-cross-tenant-replication` to `False` for security concerns

Upgrade

- `az upgrade`: Add `--allow-preview` to distinguish stable-only extension installation with preview-included extension installation

December 05, 2023

Version 2.55.0

ACR

- `az acr login`: Allow registry names with hyphen
- Fix #27487: `az acr check-health`: Fix DOCKER_PULL_ERROR when acr check-health for Mac OS

AKS

- `az aks update`: Update outbound ip description and remove limitation
- `az aks create`: Add arguments `--asg-ids` and `--allowed-host-ports`
- `az aks nodepool add/update`: Add arguments `--asg-ids` and `--allowed-host-ports`

App Service

- Fix #27189: `az webapp log tail`: Catch exception when scm connection is lost

Billing

- `az billing period list`: Fix `--top` does not work as expected
- `az billing invoice download`: Fix command does not work
- `az billing invoice list`: Fix `--period-end-date` and `--period-start-date` help message error

Compute

- `az disk create`: Add new parameter `--optimized-for-frequent-attach` to improve reliability and performance of data disks that are frequently attached
- `az disk/snapshot create`: Add new parameter `--elastic-san-resource-id` to support creating through the ARM id of elastic san volume snapshot

Containerapp

- `az containerapp ingress cors enable`: Only update arguments `--allow-headers`, `--allow-credentials`, `--allow-methods`, `--expose-methods`, `--max-age` when the value is not `None`
- `az containerapp`: Change the container-app name and container-app job name in the example to legal names

Key Vault

- `az keyvault backup start`: Add `status` in output

Monitor

- `az monitor activity-log alert`: Upgrade api-version to `2020-10-01` to include `any-of` query condition
- `az monitor activity-log alert`: Expose parameter `all-of` to enable user modifying query condition specifically

Network

- `az network private-endpoint-connection`: Add provider `Microsoft.DBforPostgreSQL/flexibleServers`
- `az network public-ip prefix`: Add parameter `--tier`

RDBMS

- `az postgres flexible-server replica create`: Add support for parameters like `--tier`, `--sku-name`, `--storage-size` during replica creation
- `az postgres flexible-server update`: Add support for custom IOPS update for flexible server using `--performance-tier`
- `az postgres flexible-server advanced-threat-protection-setting show`: Show advanced threat protection setting

- `az postgres flexible-server advanced-threat-protection-setting update`: Update advanced threat protection setting using `--state` as Enabled/Disabled

Service Connector

- `az containerapp connection create`: Enable Key Vault Reference in Container Apps

SQL

- `az instance-pool create/update`: Add optional parameter `--maintenance-configuration-id` for SQL Instance Pool resource
- `az mi create/update`: Add optional parameter `--instance-pool-name` for SQL Managed Server resource

Storage

- `az storage blob upload`: Increase `max_block_size` for append/block blobs of size $\geq 8\text{mb}$ to 8mb instead of 4mb
- `az storage blob upload`: Change default `max_connections` for append blob to 1
- `az storage file upload/upload-batch`: Fix `--allow-trailing-dot` breaking `--connection-string` usage
- Fix #27899: `az storage account create`: Run `check_name_availability()` first and throw a warning when an existing account with the name is found

November 14, 2023

Version 2.54.0

ACR

- `az acr artifact-streaming`: Add new group to include a subgroup operation and the commands create/show/update
- `az acr artifact-streaming operation`: Add new group to help the user manage the artifact streaming creation with the commands cancel/show
- `az acr login`: Add additional parameter validation to check registry name is valid
- `az acr cache`: Add optional `--resource-group` parameter
- `az acr credential-set`: Add optional `--resource-group` parameter

AKS

- [BREAKING CHANGE] `az aks create`: Make container log V2 as default and remove region dependency
- `az aks create/update`: Support custom ips/ipprefixes and managed ips being assigned to aks cluster outbound resources together
- `az aks nodepool add/update/upgrade`: Add new parameter `--drain-timeout` to slow down the upgrade
- `az aks update`: Fix bug where supportPlan can be reset to None
- `az aks nodepool add`: Fix incorrectly specified property name for option `--drain-timeout`
- `az aks create/update`: Enable Grafana support in US Government clouds
- `az aks create/update`: Update region map for default region creation with new Azure Monitor Workspace regions
- `az aks create/update`: Add cluster scope to recording rule groups created during Managed Prometheus onboarding
- `az aks update`: Add `--network-policy` to support updating the mode of a network policy

API M

- `az apim api create`: Make `--authorization-scope` parameter optional for setting auth server

App Config

- [BREAKING CHANGE] `az appconfig kv export`: Add validations to snapshot kv export to App Service
- [BREAKING CHANGE] `az appconfig kv import/export`: Add skip-features and skip-keyvault restriction for snapshots
- `az appconfig snapshot`: GA snapshot related features

App Service

- [BREAKING CHANGE] `az functionapp`: Redact key value output on keys set
- [BREAKING CHANGE] Redact output on deployment source config
- [BREAKING CHANGE] `az webapp config`: Redact webapp config connection-string and storage-account values

- [BREAKING CHANGE] `az functionapp/webapp deployment source update-token`: Redact tokens output on deployment source update-token
- [BREAKING CHANGE] `az appservice ase create`: Update the default App Service Environment to V3
- `az appservice list-locations`: Add `--hyperv-workers-enabled` parameter to get regions which support hosting web apps on windows container workers
- `az functionapp deployment source config-zip`: Add the deployer information to improve telemetry
- `az webapp up`: Fix `--logs` arg fails with an exception
- `az functionapp create`: Add new parameter `--workspace` to support creating workspace-based app insights components

ARM

- [BREAKING CHANGE] `az stack mg create`: Not supplying `--deployment-scope` will no longer default the underlying deployment to the current subscription scope, but to the mg scope of the deployment stack.
- `az stack sub create`: Add no wait support
- `az stack mg create`: Add no wait support
- `az stack group create`: Add no wait support
- `az bicep lint`: Add new command to lint a bicep file
- `az deployment group create`: Support supplemental parameters when used with `.bicepparam` parameter file
- `az deployment`: Add support for determining type of parameters whose definition uses a \$ref

ARO

- `az aro create`: Add new `--enable-preconfigured-nsg` parameter, allowing users to enable or disable preconfigured NSGs
- `az aro create`: Add network contributor to the NSG resource for the cluster SP and FP SP
- `az aro update`: Add network contributor to the NSG resource for the cluster SP and FP SP, if not already when preconfigured NSG is enabled

Backup

- `az backup vault create`: Allow updates to immutability for vaults with Managed Identity set

- `az backup vault update`: Add new command to support updating vault properties without the `--location` parameter

Batch

- `az batch keys renew`: Update help with security warning
- `az batch keys list`: Update help with security warning
- `az batch account identity show`: Update help with security warning

Compute

- [BREAKING CHANGE] Remove unversioned image aliases
- [BREAKING CHANGE] `az vm/vmss identity assign`: Remove the default value `Contributor` of parameter `--role`
- [BREAKING CHANGE] `az disk create`: Support creating disk with Gen2 and TLVM as default
- [BREAKING CHANGE] `az vm/vmss create`: Disable integrity monitoring by default
- [BREAKING CHANGE] `az disk/snapshot create`: Change the default value of `--hyper-v-generation` from `v1` to `None`
- [BREAKING CHANGE] `az vm create`: Change default value to `Standard` for LB options
- [BREAKING CHANGE] `az vmss create`: Change default value to `Standard` for LB options
- [BREAKING CHANGE] `az vmss create`: Change the default value of `--orchestration-mode` from `uniform` to `flexible`
- [BREAKING CHANGE] `az vm/vmss create`: Support Trusted Launch as default deployment option when creating vm/vmss with marketplace image
- [BREAKING CHANGE] `az vm/vmss create`: Support Trusted Launch as default deployment option when creating vm/vmss from existing disk or image
- [BREAKING CHANGE] `az vm/vmss create`: Support Trusted Launch as default deployment option when creating vm/vmss with minimal inputs
- Fix #27446: `az vm encryption enable`: Fix using incorrect client when `--key-encryption-key` is specified
- Fix #27451: `az vmss list-instances`: Fix API profile to resolve no registered resource provider found error
- `az vm create`: Format the notification message of recommendation region
- `az restore-point create`: Add new parameters `--source-os-resource`, `--os-restore-point-encryption-set` and `--os-restore-point-encryption-type` to support

encryption OS disk

- `az restore-point create`: Add new parameters `--source-data-disk-resource`, `--data-disk-restore-point-encryption-set` and `--data-disk-restore-point-encryption-type` to support encryption data disk
- `az disk create`: Add new parameter `--optimized-for-frequent-attach` to improve reliability and performance of data disks that are frequently attached
- `az disk/snapshot create`: Add new parameter `--elastic-san-resource-id` to support creating through the ARM id of elastic san volume snapshot
- `az disk create`: Revert new parameter `--optimized-for-frequent-attach`
- `az disk/snapshot create`: Revert new parameter `--elastic-san-resource-id`

Containerapp

- [BREAKING CHANGE] `az containerapp env workload-profile update`: Remove `--workload-profile-type` as it does not work in server side
- [BREAKING CHANGE] `az containerapp env create`: Update the default value of `--enable-workload-profiles` to `True`
- `az containerapp job create`: Fix AttributeError when `--trigger-type` is `None`
- `az containerapp compose create`: Fix bug where environment's resource group is not resolved from `--environment` when the input value is a resource id
- `az containerapp env workload-profile delete`: Fix issue when deleting wp for env with custom domain
- `az containerapp update`: Fix appending to `NoneType` object bug for `--secret-volume-mount`
- `az containerapp create/update`: Hide environment variables, scale rules metadata
- `az containerapp job create/update`: Hide environment variables, scale rules metadata
- `az containerapp compose create`: Fix containerapp invalid memory resource
- `az containerapp job create`: Fix problem of parsing parameters `minExecutions` and `maxExecutions` from `--yaml`

Cosmos DB

- `az cosmosdb create/update`: Add support for minimum allowed TLS version and burst capacity configuration

Eventhub

- [BREAKING CHANGE] `az eventhubs georecovery-alias update`: This command is removed.

Key Vault

- [BREAKING CHANGE] `az keyvault storage`: Remove this command group since service doesn't maintain anymore
- `az keyvault create/update-hsm`: Add `--mi-user-assigned` to support MHSM managed identity
- `az keyvault backup/restore start`: Add `--use-managed-identity` to exempt SAS token
- `az keyvault key`: Add hsm platform info in response

Monitor

- [BREAKING CHANGE] `az monitor activity-log alert create`: Change default value from `resourceGroupId` to `subscriptionId` for parameter `--scope`
- [BREAKING CHANGE] `az monitor metrics alert`: Change datetime output to be consistent with native response
- [BREAKING CHANGE] `az monitor log-analytics workspace table search-job create`: Remove `schema` wrapper for `searchResults` in api response
- `az monitor log-analytics workspace create`: Add `--identity-type` and `--user-assigned` arguments
- `az monitor log-analytics workspace update`: Add `--identity-type` and `--user-assigned` arguments
- `az monitor log-analytics workspace table`: Enable `--retention-time` to be workspace retention when setting as `-1`
- `az monitor log-analytics workspace table`: Enable `--total-retention-time` to be workspace retention when setting as `-1`
- `az monitor log-analytics workspace table search-job`: Add new command `cancel`
- `az monitor autoscale update`: Fix update failure with empty notification

MySQL

- `az mysql flexible-server gtid reset`: Remove geo-backup check

Network

- [BREAKING CHANGE] `az network public-ip create`: Change default value of `--sku` to standard
- [BREAKING CHANGE] `az network lb create`: Change default value of `--sku` to standard
- `az network private-dns record-set a add-record`: Fix record cannot be added when record set is empty
- `az network lb address-pool`: Add parameter `--sync-mode`
- `az network application-gateway listener`: Add parameter `--host-names`
- `az network private-endpoint-connection`: Add provider `Microsoft.DocumentDB/mongoClusters`
- Fix #27508: `az network private-dns zone import`: Import fails when zone already exists
- `az network virtual-appliance`: Add parameter `--identity`
- Fix #27735: `az network vnet-gateway show`: Conflict key when apply client flatten

Packaging

- [BREAKING CHANGE] Remove unnecessary packages in docker image
- Support Python 3.11
- Bump embedded Python version to 3.11.5
- Add Azure Linux docker image
- Fix #22741: `az upgrade`: This command becomes non-blocking on Windows

Profile

- `az account get-access-token`: Return `expires_on` as POSIX timestamp

RDBMS

- `az postgres flexible-server geo-restore`: Add cross subscription geo-restore support for PostgreSQL flexible server
- `az postgres flexible-server restore`: Add cross subscription restore support for PostgreSQL flexible server
- `az postgres flexible-server upgrade`: Add MVU support for PG version 15

Role

- [BREAKING CHANGE] `az role assignment create`: `--scope` is now a required argument.

- [BREAKING CHANGE] `az role assignment create`: Remove `--resource-group` argument.

Search

- `az search service create`: Add `--semantic-search` argument.

Security

- `az security pricing create`: Support subplan and extensions parameters
- `az security pricing get`: Support extensions in the return result

Service Bus

- [BREAKING CHANGE] `az servicebus georecovery-alias update` : This command is removed.
- [BREAKING CHANGE] `az servicebus migration update`: This command is removed.

Service Connector

- [BREAKING CHANGE] `az spring connection`: Remove default value of `--deployment` to support spring app connection

SignalR

- `az signalr replica create/list/show/delete`: Add replica command group for `az signalr`

SQL

- `az sql db create/update`: Add `--use-free-limit` and `--free-limit-exhaustion-behavior` to support free limit database

Storage

- `az storage file/directory`: Add `--disallow-trailing-dot`
- `az storage share list-handle/close-handle`: Add `--disallow-trailing-dot`
- `az storage file copy start/start-batch`: Add `--disallow-source-trailing-dot`

- Fix #27590: `az storage fs directory download`: Check user system PATH for azcopy and use CLI config directory for new install
- `az storage account blob-inventory-policy create`: Add support for new filter `creationTime.lastNDays` in json
- `az storage account migration start/show`: Support customer initiated migration between replication options
- [BREAKING CHANGE] `az storage container-rm update`: Remove `--default-encryption-scope` and `--deny-encryption-scope-override` as they should only be specified during create

October 24, 2023

Version 2.53.1

AKS

- Hotfix: `az aks update`: Fix bug where supportPlan can be reset to None

App Service

- [BREAKING CHANGE] Redact appsettings output on set/delete commands

September 26, 2023

Version 2.53.0

ACR

- `az acr token create`: Fix random order of repo valid actions and gateway valid actions in the help message

AKS

- `az aks update`: Add new parameter `--private-dns-zone` to support private DNS zone for AKS private cluster
- `az aks update`: Add new parameter `--disable-windows-gmsa` to support disabling Windows gMSA in an AKS cluster
- `az aks update`: Add forceupgrade settings to aks stable cli

App Config

- `az appconfig kv import/export`: Remove `skip-features` and `skip-keyvault` restriction for snapshots

App Service

- `az functionapp create`: Enable distributed tracing for non consumption apps

ARM

- `az deployment group create`: Make `--template-file` parameter optional when used with `.bicepparam` parameter file
- `az account list-locations`: Add new parameter `--include-extended-locations` to support listing extended locations

Backup

- `az backup backup-properties`: Add option for setting `--soft-delete-feature-state` to "AlwaysOn", and `--soft-delete-duration` with values between 14 to 180 (inclusive)
- `az backup vault list-soft-deleted-containers`: List all soft-deleted containers in a backup vault

Compute

- `az vm/vmss extension set`: Enable auto upgrade by default for CodeIntegrityAgent extension
- `az vm create`: Add warning message for Basic option removal
- `az vmss create`: Add warning message for Basic option removal

Containerapp

- `az containerapp`: Move `containerapp` from CLI extension to core CLI
- `az containerapp env create`: Add `--enable-workload-profiles` to specify if the environment is enabled to have workload profiles
- `az containerapp env dapr-component create`: Fix the sample link for `--yaml` parameter

Cosmos DB

- `az cosmosdb postgres`: GA Cosmos DB for PostgreSQL

MySQL

- `az mysql flexible-server replica create`: Add new parameters to support replica creation

NetAppFiles

- `az netappfiles volume`: Add new command `get-groupid-list-for-ldapuser` to Get Group Id List for LDAP User
- `az netappfiles account update`: Add parameter `--identity-type`
- `az netappfiles volume update`: Add parameter `--snapshot-dir-visible`. If enabled (true) the volume will contain a read-only snapshot directory which provides access to each of the volume's snapshots (defaults to true)

Network

- `az network virtual-appliance`: Add parameter `--additional-nics`
- `az network vnet subnet`: Add parameter `--default-outbound-access`
- `az network public-ip create`: Add warning message for Basic option removal
- `az network lb create`: Add warning message for Basic option removal

RDBMS

- `az postgres flexible-server create/update`: Add capability to enable/disable storage auto-grow during creation and update

Service Connector

- `az spring connection`: Add deprecated message for `--deployment` breaking change
- `az webapp connection`: Add new parameter `--slot` to support webapp slot connection

SQL

- `az sql failover-group set-primary`: Add parameter `--try-planned-before-forced-failover` to support hybrid geo-failover

Storage

- Fix #26732: `az storage blob copy start-batch`: Add `--rehydrate-priority` to batch copy
- Fix #27052: `az storage blob delete-batch`: Use utc as default timezone to remove "Datetime with no tzinfo will be considered UTC." warning

September 05, 2023

Version 2.52.0

AKS

- `az aks create/update`: Add new parameter `--enable-vpa` to support enabling vertical pod autoscaler for cluster
- `az aks update`: Add new parameter `--network-dataplane` to specify the network dataplane used in the Kubernetes cluster
- `az aks create/update`: Add new parameter `--node-os-upgrade-channel` to specify which OS on your nodes is updated
- `az aks update`: Retain value in network profile in mc object only when decorator is in update mode
- `az aks update`: Outbound ip/outbound ipprefix and managed outbound ip should be mutually exclusive

App Config

- `az appconfig kv import`: Add new parameter `--import-mode` to specify whether to overwrite already existing key-values or ignore matching keys
- `az appconfig kv export`: Add new parameter `--snapshot` to support exporting all key values from a snapshot of the source configuration
- `az appconfig kv import`: Add new parameter `--src-snapshot` to support importing all key values from a snapshot of the source configuration

App Service

- Fix #26736: `az logicapp create`: Add `--runtime-version` and `--functions-version` optional parameters
- `az webapp config connection-string set`: Allow users to use json file to set the connection string

ARM

- Fix #26112: `az deployment group create`: Fix the warning log `mode is not a known attribute of class TemplateLink`
- `az bicep build-params`: Support generating `parameters.json` file from the given `bicepparam` file with the `--file` argument
- `az bicep decompile-params`: Support generating `parameters.bicepparam` file from the given `parameters.json` file with the `--file` argument
- `az bicep generate-params`: Support generating `main.parameters.json` with the parameters that doesn't have default values in the given `.bicep` file
- `az bicep generate-params`: Add new parameter `--output-format` to support generating parameter file in `bicepparam` and `json` formats
- `az bicep generate-params`: Add new parameter `--include-params` to support generating parameter file with all the parameters in the given `bicep` file, or with only parameters that doesn't have default values in the given `bicep` file

ARO

- `az aro create`: Add new `--outbound-type` parameter, allowing users to select "Loadbalancer" (default) or "UserDefinedRouting"
- `az aro create`: Perform pre-flight validation of prerequisite permissions before creation
- `az aro validate`: New command to perform explicit validation of prerequisite permissions

Backup

- `az backup restore restore-azurefileshare`: Add `--target-rg-name` parameter to specify the resource group of the destination storage account

Batch

- `az batch`: Fix batch cloud console authentication issue

Cognitive Services

- `az cognitiveservices account deployment create`: Add `--model-source` parameter

Compute

- `az vmss create/update`: Add `--enable-hibernation` parameter to enable hibernation capability on VMSS
- `az vmss update`: Add `--security-type` parameter to enable Trusted Launch on existing VMSS
- `az vmss deallocate`: Add `--hibernate` parameter to support hibernating a VM while deallocating
- `az ppg update`: Add new parameter `--type` to support setting proximity placement group type

Cosmos DB

- `az cosmosdb restore`: Support enabling/disabling public network access

Key Vault

- Fix #27220: `az keyvault certificate import`: Fix invalid policy issue when no `content_type` provided
- `az keyvault storage`: Announce deprecation since keyvault service doesn't maintain this since long ago

MySQL

- `az mysql flexible-server parameter set-batch`: Add new command to support updating multiple parameters
- `az mysql flexible-server export create`: Add Export Backup CLI implementation

Network

- `az network private-endpoint-connection`: Add provider `Microsoft.EventGrid/namespaces` and `Microsoft.EventGrid/partnerNamespaces`
- Fix #27066: `az network vnet list`: Fix -o table cannot be used
- `az network express-route port delete`: Add confirmation while deleting

- `az network application-gateway waf-policy custom-rule`: Add an example of using `--group-by-user-session`
- `az network express-route update`: Fix `properties.ServiceProviderProperties` unexpected null
- Fix #26730: `az network public-ip update`: `--ip-tags` cannot be correctly parsed
- `az network application-gateway waf-policy managed-rule rule-set`: Support Microsoft_BotManagerRuleSet version 1.0
- `az network vnet peering create`: Mark `--remote-vnet` as required

Redis

- `az redis update`: Fix public network access default value issue

Storage

- `az storage file upload-batch`: Allow uploading files in parallel to improve performance
- Fix #27202: `az storage entity insert`: Fix case when using sas token with only `add` permission

Upgrade

- `az upgrade`: Support upgrading with 64-bit MSI

August 01, 2023

Version 2.51.0

AKS

- `az aks nodepool snapshot update`: Add aks nodepool snapshot update command
- `az aks create`: Add new parameter `--k8s-support-plan` to support LTS onboarding, also add new tier enum `premium`
- `az aks update`: Support enabling/disabling LTS via new parameter `--k8s-support-plan`
- `az aks create`: Add node taint support when create cluster use `az aks` command
- `az aks update`: Add update node taint support on cluster level use `az aks` command

- `az aks enable-addons`: Fix the default value of option `--enable-msi-auth-for-monitoring` being overwritten to `False` when specified
- `az aks update`: Add new parameter `--outbound-type` to support cluster outbound type.
- `az aks maintenanceconfiguration list`: Add new command to list all maintenance windows in a cluster
- `az aks maintenanceconfiguration show`: Add new command to display a specific maintenance window of a cluster
- `az aks maintenanceconfiguration add`: Add new command to add a new maintenance window configuration for a cluster
- `az aks maintenanceconfiguration update`: Add new command to update an existing maintenance window configuration of a cluster
- `az aks maintenanceconfiguration delete`: Add new command to delete an existing maintenance window configuration of a cluster
- `az aks update`: Fix aks network profile update error

App Config

- `az appconfig kv delete/set/set-keyvault`: Add key validations for null or empty space keys
- `az appconfig kv export/import/restore`: Update key-value diffing and preview
- `az appconfig snapshot`: Remove status code property from snapshot object
- `az appconfig snapshot list`: Use enums for status parameter

App Service

- Fix #26214: `az webapp show`: Fix the bug caused by missing leading slash causes web app and plan commands to fail for s-clouds
- Fix #26214: `az appservice plan show`: Fix the bug caused by missing leading slash causes web app and plan commands to fail for s-clouds
- Fix #26601: `az functionapp create`: Throw error for consumption function app created with vnet
- Fix #21133: `az webapp/functionapp config ssl bind/unbind`: Search for matching certificates in the subscription by App Service Plan Id

ARM

- `az stack`: Fix the bug that the required `--deny-settings-mode` parameter should not return None (should be a string)

- `az stack`: Fix the bug that the `--deny-settings-excluded-principals` parameter was accidentally reset

Batch

- `az batch job/pool all-statistics`: Remove no longer worked commands
- `az batch pool create`: Add new parameter `--enable-accelerated-networking` to determine whether this pool should enable accelerated networking

Cognitive Services

- `az cognitiveservices account deployment create`: Add `--sku-name` and `--sku-capacity` parameters
- `az cognitiveservices usage`: Add new command `list`
- `az cognitiveservices model`: Add new command `list`

Compute

- `az vm/vmss create`: Enable auto upgrading of guest attestation extension by default for Trusted Launch enabled VMs and VMSS
- `az vm/vmss create`: Add new parameter `--disable-integrity-monitoring-autoupgrade` to support disabling auto upgrading of guest attestation extension for Trusted Launch enabled VMs and VMSS
- `az sig image-version undelete`: Add new command to support softdeleted image recovery
- `az vm/vmss/disk create`: Add new option `Standard` for `--security-type` for backward compatibility
- `az sig image-definition create`: Add new option `Standard` for `--security-type` for backward compatibility

Cosmos DB

- `az cosmosdb restore`: Add `--assign-identity` and `--default-identity` to allow PITR restoring with identity
- `az cosmosdb postgres`: Add new command groups to support Cosmos DB for PostgreSQL

Key Vault

- `az keyvault restore start`: Add `--key-name` to support selective key restoring
- `az keyvault key sign/verify`: Add new commands to support signing with keyvault key and verify the signature

MySQL

- `az mysql flexible-server ad-admin set`: Enable AAD for replica

Network

- `az network nic create/update`: Add parameters `--auxiliary-mode` and `--auxiliary-sku` to support setting auxiliary mode and sku
- `az network public-ip`: Add parameter `--dns-name-scope` to specify different options
- `az network private-endpoint-connection`: Add provider `Microsoft.ElasticSan/elasticSans`

Packaging

- Drop Python 3.7 support
- Support x86 and x64 MSI builds

Resource

- `az resource invoke-action`: Add new parameter `--no-wait` to support not waiting the long-running operation to finish

Role

- `az ad sp create-for-rbac`: Add alias `--json-auth` for `--sdk-auth`

Service Connector

- `az functionapp connection`: Add new command group to support service connector on Function App
- `az spring connection`: Enable new auth types for Spring Boot and Cosmos SQL connection

SQL

- `az sql mi start/stop/start-stop-schedule`: Add SQL MI manual and scheduled start stop

Storage

- `az storage container-rm update`: `--default-encryption-scope` and `--deny-encryption-scope-override` should not be specified during updating
- Fix #22704: `az storage account create`: `--encryption-key-type-for-queue` and `--encryption-key-type-for-table` no longer remove other settings
- Fix #26587: `az storage file upload`: Add `--file-url` to support supplying the url instead of share/file name

July 04, 2023

Version 2.50.0

AKS

- `az aks get-versions`: Both json payload and table format changed
- `az aks create`: Add condition to disable `--enable-msi-auth-for-monitoring` for service principle

APIM

- `az apim graphqlapi resolver create`: Add new command to create gql api resolver
- `az apim graphqlapi resolver show`: Add new command to show gql api resolver
- `az apim graphqlapi resolver list`: Add new command to show gql api resolver list
- `az apim graphqlapi resolver delete`: Add new command to delete gql api resolver
- `az apim graphqlapi resolver policy create`: Add new command to create resolver policy
- `az apim graphqlapi resolver policy show`: Add new command to show resolver policy
- `az apim graphqlapi resolver policy list`: Add new command to list resolver policies
- `az apim graphqlapi resolver policy delete`: Add new command to delete a policy

App Config

- `az appconfig feature`: Improve error handling for invalid feature flags
- `az appconfig snapshot create`: Add new command to support creating a snapshot
- `az appconfig snapshot show`: Add new command to support showing the properties of an app configuration snapshot
- `az appconfig snapshot list`: Add new command to support listing snapshots of a given app configuration
- `az appconfig snapshot archive`: Add new command to support archiving a snapshot
- `az appconfig snapshot recover`: Add new command to support recovering an archived snapshot

App Service

- Fix #21168: `az webapp deploy`: Call OneDeploy through ARM proxy if `--src-url` is provided
- Fix #26647: `az webapp show`: Remove duplicate IPs from outbound addresses
- Fix #25497: `az webapp deploy`: Fix extension parsing if `src-path` has multiple '.'s

ARM

- `az managedapp definition create/update`: Add new parameter `--deployment-mode` to support setting deployment policy
- `az resource move`: Add help example for moving multiple resources
- `az stack`: Add new command group to support deployment stacks
- `az stack mg`: Add new command group to manage deployment stack at management group scope
- `az stack sub`: Add new command group to manage deployment stack at subscription scope
- `az stack group`: Add new command group to manage deployment stack at resource group scope

ARO

- `az aro get-admin-kubeconfig`: Add new command to download an admin kubeconfig for a created ARO cluster

Backup

- `az backup vault create`: Add parameter `--cross-subscription-restore-state` to set the CSR state of the vault at the time of creation as well as updating
- `az backup recoveryconfig show`: Add parameter `--target-subscription-id` to provide the target subscription as the input while triggering cross subscription restore for SQL or HANA workloads
- `az backup protection backup-now`: Allow `--enable-compression` to be set to `true` for SAPHANA Workloads
- `az backup recoveryconfig show`: Add new parameter `--target-instance-name` to specify the target instance name for the restore operation

Compute

- `az vmss update`: Add new parameter `--custom-data` to support updating custom data
- `az image builder optimizer`: Add subgroup to manage image template optimizer
- `az image builder create`: Add parameter `--validator` to specify the type of validation to be used on the Image
- `az vm update`: Add parameter `--security-type` to support VM Gen2 to Trusted Launch conversion
- `az sig image-definition create`: Add examples for TrustedLaunchSupported and TrustedLaunchAndConfidentialVmSupported
- `az capacity`: Fix short summaries for groups
- Fix #26516: `az vm create`: Fix warning log for public IP even when no public IP is being created

Eventhub

- `az eventhubs eventhub`: Enum value for `cleanup_policy` change to `compact` from `compaction`
- `az eventhubs namespace list`: Support list command without mandatory `resource_group` parameter
- `az eventhubs eventhub create/update`: Event Hubs Capture MSI feature added to eventhub entity

IoT

- `az iot hub route`: Hide the deprecated command, please use `az iot hub message-route` instead of it.

- `az iot hub routing-endpoint`: Hide the deprecated command, please use `az iot hub message-endpoint` instead of it.

Key Vault

- Fix #26527: `az keyvault certificate show`: Show `policy.x509CertificateProperties.subjectAlternativeNames` correctly

Monitor

- `az monitor metrics alert create`: Add `()` into `--condition` grammar

MySQL

- `az mysql flexible-server import create`: Add new command to facilitate migrations from mysql single to flexible servers
- `az mysql flexible-server restore/geo-restore/replica create`: Support `--tags`

NetAppFiles

- `az netappfiles volume replication resume`: Add warning on action to re-sync replication volumes that if destination volume has quota rules they will be overwritten by the source volumes quota rules.

Network

- `az network dns zone import`: Fix alias records cannot be imported
- Fix #26438: `az network vnet peering sync`: Doesn't work in cross-tenant scenario
- `az network application-gateway waf-policy policy-setting update`: Add support for log scrubbing
- `az network application-gateway waf-policy policy-setting update`: Add support for inspection limit
- `az network application-gateway waf-policy custom-rule`: Support rate limit in WAF policy
- Fix #24695: `az network traffic-manager`: Add command context
- Fix #26638: `az network traffic-manager endpoint`: Declare `--min-child-endpoints`, `--min-child-ipv4` and `--min-child-ipv6` as integer type
- `az network dns`: Support DNSSEC configuration and DS/TLSA record set

Packaging

- Add Debian Bookworm support

RDBMS

- `az postgres flexible-server migration update`: Remove unsupported update parameters `--db-names` and `--overwrite-dbs`
- `az postgres flexible-server migration create`: Add support for tags and location using `--tags` and `--location`
- `az postgres flexible-server revive-dropped`: Add support to revive a dropped PostgreSQL flexible server
- `az postgres flexible-server create`: Add support to create PostgreSQL flexible server with data encryption enabled for geo-backup enabled server by passing parameters `--geo-redundant-backup`, `--backup-key` and `--backup-identity`
- `az postgres flexible-server show-connection-string`: Add support to pass `--pg-bouncer` in connection strings for cmd and programming languages with PgBouncer enabled for PostgreSQL flexible server. Updated connection strings to show port as well as database
- `az postgres flexible-server update`: Add support for parameter `--private-dns-zone` during update operation, to update private DNS zone for a VNET enabled PostgreSQL flexible server

Service Bus

- `az servicebus namespace list`: Support list command without mandatory `resource_group` parameter

Service Fabric

- `az sf managed-cluster network-security-rule add`: Add network security rule to managed cluster

SQL

- `az sql midb move/copy`: Add new commands for Managed Database Move/Copy feature

SQL VM

- Fix #2442969: `az sql vm enable-azure-ad-auth/validate-azure-ad-auth`: Workaround Graph API bug by using client side filtering upon failure
- `az sql vm update`: Add configuration options for new SQL Assessment prerequisites MMA->AMA migration

Storage

- `az storage blob upload(-batch)/set-tier/copy start(-batch)`: Cold Tier GA, add new tier type `--tier cold`
- `az storage blob download-batch`: When matching pattern, list blobs with prefix to reduce the number of list calls
- Fix #26673: `az storage account or-policy create`: Now throw server error that was previous silently ignored.

Synapse

- `az synapse workspace create/update`: Support workspace encryption and user-assignment management identity

May 23, 2023

Version 2.49.0

ACR

- `az acr create`: Remove `Classic` from `sku` option

AKS

- [BREAKING CHANGE] `az aks create`: Specify `--pod-cidr` with Azure CNI will return an error instead of logging a warning when not use `overlay` mode
- [BREAKING CHANGE] `az aks create`: Change the default value of `--enable-msi-auth-for-monitoring` to true and add check for airgap clouds
- `az aks update`: Support updating user assigned control plane identity for parameter `--assign-identity`
- `az aks install-cli`: Add validation for installation path and update help message for parameters `--install-location` and `--kubelogin-install-location`
- Fix #26353: `az aks install-cli`: Fix incorrect architecture detection on Darwin/arm64

- `az aks create/update`: Add parameter `--enable-azure-monitor-metrics` to enable managed prometheus (Azure Monitor Metrics Addon)
- `az aks create/update`: Add parameter `--azure-monitor-workspace-resource-id` to store metrics for the managed prometheus addon
- `az aks create/update`: Add parameter `--grafana-resource-id` to link the Azure Monitor Workspace with a Grafana instance for viewing metrics and dashboards
- `az aks create/update`: Add parameter `--enable-windows-recording-rules` to enable windows recording rule groups on the Azure Monitor Workspace (by default they get created but are disabled)
- `az aks create/update`: Add parameter `--ksm-metric-labels-allow-list` to support the additional Kubernetes label keys that will be used in the resource's labels metric
- `az aks create/update`: Add parameter `--ksm-metric-annotations-allow-list` to support the Kubernetes annotations keys that will be used in the resource's labels metric
- `az aks update`: Add parameter `--disable-azure-monitor-metrics` to disable the Azure Monitor Metrics addon
- `az aks create` and `az aks nodepool add`: Add warning message when specifying `-os-sku` to Mariner or CBLMariner

App Config

- [BREAKING CHANGE] `az appconfig feature`: Update feature name validation to disallow the colon character
- [BREAKING CHANGE] `az appconfig kv import`: Update feature name validation. Invalid feature flags will be skipped during import
- [BREAKING CHANGE] `az appconfig`: Update default connection string resolution logic

App Service

- `az functionapp create`: Add new parameter `--min-replicas` and `--max-replicas` to support minimum and maximum replicas
- `az functionapp create`: Add new parameter `--registry-server` to support Centauri function app
- `az functionapp create`: Update the default image to `mcr.microsoft.com` for Centauri
- Fix #26445: `az webapp deploy`: Fix deployment failing with HTTP 400

ARM

- Fix #26216: `az bicep format`: Fix the `TypeError` `expected str, bytes or os.PathLike object, not bool`
- Fix #26256: `az bicep publish/restore/generate-params`: Fix version checks without bicep installed
- `az bicep publish`: Add new parameter `--force` to allow overwriting existing module
- Fix #26352: `az ts create`: Fix for the `TypeError` `string indices must be integers`

Backup

- `az backup`: Add support for HANA HSR workload

Compute

- `az vm create`: Support new license type `UBUNTU_PRO` and `UBUNTU`
- `az vm extension set`: Enable auto-upgrade by default for GuestAttestation extension
- `az image builder trigger`: Add subgroup to manage image builder template trigger
- `az image builder output versioning`: Add subgroup to manage image builder template output versioning
- `az image builder output add`: Add parameter `--versioning` to support describing how to generate new x.y.z version number for distribution
- `az image builder output add`: Add parameter `--vhd-uri` to support specifying storage uri for the distributed VHD blob

Container

- `az container create`: Add new parameters for container security context for confidential ContainerGroupSku

Cosmos DB

- [BREAKING CHANGE] `az cosmosdb create/update`: Rename `--enable-public-network true/false` to `--public-network-access ENABLED/DISABLED/SECUREDBYPERIMETER`

- `az cosmosdb create/update`: Add `--continuous-tier` to support continuous backup tier
- `az cosmosdb create/update`: Enable Partition Merge feature for CosmosDB

Eventhub

- [BREAKING CHANGE] `az eventhubs namespace network-rule`: This command group is removed and replaced by `az eventhubs namespace network-rule-set`
- [BREAKING CHANGE] `az eventhubs namespace network-rule add`: This command is removed and replaced by `az eventhubs namespace network-rule-set ip-rule/virtual-network-rule add`
- [BREAKING CHANGE] `az eventhubs namespace network-rule remove`: This command is removed and replaced by `az eventhubs namespace network-rule-set ip-rule/virtual-network-rule remove`
- [BREAKING CHANGE] `az eventhubs eventhub create/update`: Remove `--message-retention` parameter, it is replaced by `--retention-time-in-hours`
- [BREAKING CHANGE] `az eventhubs namespace application-group policy remove`: Rename `--throttling-policy-config` to `--policy` and remove `metric-id` and `rate-limit-threshold` properties in it
- `az eventhubs eventhub create/update`: Add `--cleanup-policy`, `--retention-time-in-hours` and `--tombstone-retention-time-in-hours` to support `Retention-Description` feature

IoT

- `az iot hub create/update/delete`: Fix poller issues

Key Vault

- [BREAKING CHANGE] `az keyvault create`: `--retention-days` becomes required for MHSM creation
- [BREAKING CHANGE] `az keyvault backup start`: The output will only contain `folderUrl`
- [BREAKING CHANGE] `az keyvault restore start`: Nothing will return for successful run
- [BREAKING CHANGE] `az keyvault role assignment delete`: Nothing will return for successful run

- [BREAKING CHANGE] `az keyvault certificate show/set-attributes/import`: No longer return `x509CertificateProperties.basicConstraints`, `pending`
- [BREAKING CHANGE] `az keyvault certificate contact delete`: Return an empty list instead of the deleted contact for consistency if the operation would remove the last contact
- [BREAKING CHANGE] `az keyvault certificate issuer create`: `organizationDetails.zip` is no longer returned by service, use 0 as the default
- `az keyvault security-domain upload`: Fix sd warpping keys with passwords
- `az keyvault setting`: New command group to manage MHSM settings

Monitor

- `az monitor`: Add new subgroup `account` to support managing monitor workspace
- `az monitor log-analytics workspace table create/update`: Max `--total-retention-time` changed from 2555 to 2556

NetAppFiles

- [BREAKING CHANGE] `az netappfiles volume create`: Remove optional parameter `--vault-id` as this is not longer needed
- [BREAKING CHANGE] `az netappfiles vault list`: Remove command `vault list` as this is not longer needed
- `az netappfiles account create`: Add optional parameter `--identity-type`
- `az netappfiles account ad add`: Add optional parameter `--preferred-servers-for-ldap-client`
- `az netappfiles volume create`: Add optional parameter `--is-large-volume`
- `az netappfiles volume account create`: Add optional parameter `--identity-type`
- `az netappfiles volume quota-rule update`: Add optional parameter `--tags`
- `az netappfiles volume`: Add new command `break-file-locks` to break all the file locks on a volume

Network

- [BREAKING CHANGE] `az network cross-region-lb rule`: Remove parameters `--enable-tcp-reset` and `--idle-timeout`
- [BREAKING CHANGE] `az network application-gateway http-settings update`: Use `null` instead of `""` to detach

- [BREAKING CHANGE] `az network application-gateway settings update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network application-gateway url-path-map update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network nic update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network nic ip-config update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network nsg rule update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network vnet update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network vnet subnet update`: Use `null` instead of `""` to detach
- [BREAKING CHANGE] `az network application-gateway client-cert remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network application-gateway ssl-profile remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network private-endpoint dns-zone-group remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network private-endpoint ip-config remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network private-endpoint asg remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network nic ip-config address-pool remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network nic ip-config inbound-nat-rule remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network lb address-pool tunnel-interface remove`: Deprecate the output of command
- [BREAKING CHANGE] `az network cross-region-lb address-pool address remove`: Deprecate the output of command
- `az network private-endpoint-connection`: Add provider
`Microsoft.HardwareSecurityModules/cloudHsmClusters`
- Fix #26248: `az network dns record-set cname set-record`: Declare TTL as integer type
- Fix #26326: `az network vnet subnet update`: `--nat-gateway` cannot be set to null
- Fix #26318: `az network vnet subnet create`: `--nsg` and `--route-table` cannot be used as name from Azure Stack

RDBMS

- [BREAKING CHANGE] `az mysql/postgres flexible-server create/update`: Deprecate `Enabled` for `--high-availability` argument
- `az mysql flexible-server restore/georestore/replica create`: Add `--public-access` parameter for restore, replication and georestore

Service Bus

- [BREAKING CHANGE] `az servicebus georecovery-alias fail-over`: Remove `--parameters` argument
- [BREAKING CHANGE] `az servicebus namespace network-rule`: This command group is removed and replaced by `az servicebus namespace network-rule-set`
- [BREAKING CHANGE] `az servicebus namespace network-rule add`: This command is removed and replaced by `az servicebus namespace network-rule-set ip-rule/virtual-network-rule add`
- [BREAKING CHANGE] `az servicebus namespace network-rule remove`: This is removed and replaced by `az servicebus namespace network-rule-set ip-rule/virtual-network-rule remove`
- [BREAKING CHANGE] `az servicebus queue update`: Remove deprecated parameters `--enable-partitioning`, `--enable-session` and `--duplicate-detection`

Service Connector

- `az spring connection create`: Enable user-assigned managed identity for spring apps and deprecate Postgresql single server

SQL

- [BREAKING CHANGE] `az sql mi link create`: Remove `--replication-mode` argument
- `az sql elastic-pool`: Add `--preferred-enclave-type` argument
- `az sql mi link update`: Fix update command to use PATCH api

Storage

- `az storage account blob-service-properties cors-rule`: Add new command group to manage blob cors rules

April 25, 2023

Version 2.48.1

App Service

- Hotfix: Use basic auth with SCM sites if supported, else use AAD auth

April 25, 2023

Version 2.48.0

App Service

- Fix #25642: `az functionapp deployment user show`: Fix the `AttributeError`
`'function' object has no attribute 'metadata'`
- `az staticwebapp appsettings`: Add `--environment-name` parameter to allow app setting operation on preview environments
- `az functionapp create`: Update 'kind' attribute for Centauri function apps

Compute

- Fix #26118: `az vm list-ip-addresses`: Fix the `KeyError` when attributes missing in public IP address
- Fix #26164: `az vmss update`: Fix unexpected error while running the update instance protection command on VMSS flex instances
- Fix #26185: `az sig update`: Fix issues that `is_soft_delete_enabled` may not exist
- `az vm host resize`: Add new command to support resizing dedicated host
- `az vm host list-resize-options`: Add new command to support getting possible resize options

DMS

- `az dms project task create`: Add support for database schema migration

Eventhub

- `az eventhubs namespace application-group policy remove`: Add upcoming breaking change notification

Network

- `az network nic update`: Add `--ip-configurations` to support shorthand syntax
- `az network public-ip prefix create`: Add parameter `--ip-tags`
- `az network cross-region-lb rule create`: Set default value for `--enable-tcp-reset` and `--idle-timeout`

RDBMS

- `az mysql flexible-server create/update/gtid reset`: Add GTID reset and fix public-access

SQL

- `az sql midb ledger-digest-uploads`: Support SQL Ledger
- `az sql mi server-configuration-option`: New command group to manage server configuration options

SQL VM

- `az sql vm enable-azure-ad-auth/validate-azure-ad-auth`: Single mode improvement

Storage

- `az storage file/directory`: Add `--auth-mode login` and `--backup-intent` to support OAuth
- `az storage blob sync`: Add positional argument `extra_options` to pass through options to `azcopy`

April 04, 2023

Version 2.47.0

AKS

- `az aks create/update`: Add `--tier` argument will specify the sku tier that customer wants

- `az aks nodepool operation-abort`: Add new command to support aborting last running operation on nodepool
- `az aks operation-abort`: Add new command to support aborting last running operation on managed cluster
- `az aks create`: Raise a ClientRequestError when creating the same cluster again
- `az aks create/update`: Add new parameter `--enable-image-cleaner` to enable Image Cleaner service
- `az aks create/update`: Add new parameter `--image-cleaner-interval-hours` to set Image Cleaner scanning interval
- `az aks create`: Add new parameter `--network-plugin-mode` to support creating Azure CNI Overlay clusters
- `az aks create/update`: Add new parameter `--enable-workload-identity` to support enabling workload identity addon
- `az aks create`: Add new parameter `--network-dataplane` to support creating Cilium clusters
- `az aks update`: Add parameter `--network-plugin-mode` to update the mode of a network plugin
- `az aks update`: Add parameter `--pod-cidr` to update the pod CIDR for a cluster

App Config

- `az appconfig import/export`: Add warning log info to output even when `--yes` flag is set
- `az appconfig kv import`: Ensure the case of imported boolean values does not change for string conversion from file

App Service

- Fix #25375: `az functionapp deployment source config-zip`: Fix the `Could not find a 'AzureWebJobsStorage' application setting` error
- Fix #25876: `az webapp config ssl import`: Fix the `UnboundLocalError local variable 'cert_name' referenced before assignment`
- `az functionapp create`: Support container app deployments
- `az functionapp delete`: Add a validation to check whether Azure Functions is not in the Azure Container app environments

ARM

- `az deployment group create`: Support deployment with bicepparam files

- `az resource patch`: Add new command to support updating resource by PATCH request
- Fix #25706: `az bicep format`: Fix the `TypeError ensure_bicep_installation() missing 1 required positional argument 'cli_ctx'`
- Fix #25715: `az bicep install/upgrade`: Fix the `configparser.NoSectionError: No section: 'bicep'`

Compute

- `az vm reimagine`: Add new command to support reimaging a virtual machine
- `az vm/vmss create`: Deprecate image alias `UbuntuLTS` and `Win2008R2SP1`. Please use the image alias including the version of the distribution you want to use. For example: Please use `Ubuntu2204` instead of `UbuntuLTS`

Cosmos DB

- `az cosmosdb identity assign`: Allow refreshing user assigned identities if they're reassigned to an account

Extension

- `az extension add`: Add actionable message for extension not found error

Key Vault

- `az keyvault region`: GA MHSM region commands

Monitor

- `az monitor activity-log alert`: Adjust help message

NetAppFiles

- `az netappfiles volume update`: Patch assign snapshotpolicyID

Network

- [BREAKING CHANGE] `az network`: Clean up irrelevant commands in azure-stack profiles.

- [BREAKING CHANGE] `az network application-gateway waf-policy custom-rule`: Rename output property `applicationGatewayIpConfigurations` to `applicationGatewayIPConfigurations` to keep consistent with the name in API
- `az network routeserver create/update`: Add parameter `--hub-routing-preference`
- Drop `azure-mgmt-network` SDK
- Fix #25784: `az network private-link-service update`: `--lb-frontend-ip-configs` cannot be used

RDBMS

- [BREAKING CHANGE] `az postgres flexible-server replica create`: Fix the behavior of AZ selection in case zone is not passed as parameter
- Fix #368903181: Fix zone selection during creation of replica
- `az mysql flexible-server restore/geo-restore`: Add parameters to enhance PITR
- `az mysql flexible-server replica create`: Add parameters to support cross region paired vnet

Service Bus

- `az servicebus namespace/topic/queue authorization-rule keys renew/list`: Add `-n` option for `--authorization-rule-name` to create auth rule

SQL

- `az sql server refresh-external-governance-status`: New command for refreshing external governance status
- `az sql db geo/ltr-backup restore`: Add more parameters to geo restore and ltr restore

Storage

- `az storage blob copy start`: Fix auth issue when providing source uri containing sas token
- `az storage container/blob list`: Fix MemoryError when service returns less num than requested
- `az storage account create`: GA partition DNS account support

Synapse

- `az synapse spark pool create/update`: Update `--node-size-family` and `--node-size` allowed values

March 07, 2023

Version 2.46.0

ACR

- `az acr token\scope-map`: Tokens and Scope-Maps are Generally Available
- `az acr manifest`: Support oci image index
- `az acr cache`: Add new command group to manage cache rules
- `az acr credential-set`: Add new command group to manage credential sets
- Fix #24886: `az acr`: Improve the 429 error handling for CONNECTIVITY_REFRESH_TOKEN_ERROR

AKS

- `az aks check-acr`: Fix mariner node missing cert
- Fix #25521: `az aks nodepool upgrade`: Fix the crashes when the version passed in through the parameter `--kubernetes-version` is the same as the cluster version
- Fix #25530: `az aks nodepool upgrade`: Fix agent pool property name used for fetching current k8s version
- `az aks create`: Add new parameter `--pod-cidrs` for setting the IP ranges used to allocate IPs to pods
- `az aks create`: Add new parameter `--service-cidrs` for setting the K8s service IPs
- `az aks create`: Add new parameter `--ip-families` for setting the IP types that should be used in a cluster (IPv4 or IPv6)
- `az aks create`: Add new parameter `--load-balancer-managed-outbound-ipv6-count` for setting the number of IPv6 outbound IPs that AKS should managed for a cluster with IPv6 enabled
- `az aks update`: Support changing the load balancer managed outbound IPv6 count property
- Fix #22321: `az aks get-credentials`: Fix path separator for Windows when finding `kubeconfig_path`

APIM

- Fix #25168: `az apim update`: Fix the bug that `--public-network-access` doesn't work to disable public network access

App Service

- `az webapp deleted restore`: Add new parameter `--target-app-svc-plan` to support setting app service plan for new azure web app
- Fix #14729: `az webapp config ssl upload`: Refine error handling for `OpenSSL.crypto.Error` when obtaining the certificate's thumbprint failed
- `az functionapp create`: Add new parameter `--environment` to support setting the name of container app environment
- `az webapp config ssl bind/unbind`: Allow user to specify hostname to (un)bind with `--hostname`
- `az webapp config ssl create/upload/import`: Allow user to specify certificate name with `--certificate-name`
- `az functionapp create`: Add new parameter `--environment` to support setting the name of container app environment

ARM

- `az bicep publish`: Update command to support new optional parameter `--documentationUri`
- Fix #25510: `az bicep`: Set `bicep.use_binary_from_path` to false when installed using Azure CLI
- `az bicep format`: Add new command to support formatting a Bicep file

ARO

- `az aro create`: Rename the create `install-version` parameter to `version`

Compute

- `az vmss reimagine`: Fix the bug that all instances will be reimaged after using `--instance-id` and add new parameter `--instance-ids` to replace `--instance-id`
- `az vm create`: Support recommending more suitable regions through warning log when creating VM
- `az vm/vmss identity assign`: Add warning and modify help message for `--role`: Please note that the default value of `--role` will be removed in the breaking

change release of the fall of 2023, so please specify `--role` and `--scope` at the same time when assigning a role to the managed identity.

Container

- `az container create`: Add new parameters `--priority`, `--sku` and `--cce-policy` for container group

Cosmos DB

- `az cosmosdb container create`: GA Client Side Encryption feature
- `az cosmosdb container update`: Fix updates failure for containers with client encryption policy
- `az cosmosdb restore`: Add `--gremlin-databases-to-restore`, `--tables-to-restore` parameters to support restore of gremlin and table accounts
- `az cosmosdb gremlin restorable-resource list`: New command that list restorable gremlin resources
- `az cosmosdb gremlin restorable-database list`: New command that list restorable gremlin databases
- `az cosmosdb gremlin restorable-graph list`: New command that list restorable graphs under a gremlin database
- `az cosmosdb gremlin retrieve-latest-backup-time`: New command that retrieve latest backup time for a graph under a database
- `az cosmosdb table restorable-resource list`: New command that list restorable table resources
- `az cosmosdb table restorable-table list`: New command that list restorable tables
- `az cosmosdb table retrieve-latest-backup-time`: New command that retrieve latest backup time for a table

Deployment Manager

- [BREAKING CHANGE] `az deploymentmanager`: Remove command module since it is no longer maintained by service

Key Vault

- `az keyvault create/update-hsm`: Add `--public-network-access` for MHSM creating or updating

Network

- [BREAKING CHANGE] `az network lb address-pool create/update`: Replace preview argument `--backend-addresses-config-file` and `--config-file` by `--backend-addresses` which supports Json, files and shorthand syntax formats
- [BREAKING CHANGE] `az network lb address-pool`: Output properties `privateIpAddress`, `privateIpAddressVersion`, `privateIpAllocationMethod`, `publicIpAddress` and `publicIpPrefix` are renamed by `privateIPAddress`, `privateIPAddressVersion`, `privateIPAllocationMethod`, `publicIPAddress` and `publicIPPrefix` to keep consistent with the name in API
- [BREAKING CHANGE] `az network cross-region-lb probe`: Deprecate command group as probes are not supported for global load balancer
- [BREAKING CHANGE] `az network nic create/update`: Rename output property `enableIpForwarding` to `enableIPForwarding` to keep consistent with the name in API
- [BREAKING CHANGE] `az network nic create/update`: Rename output property `privateIpAllocationMethod` to `privateIPAllocationMethod` to keep consistent with the name in API
- [BREAKING CHANGE] `az network nic create/update`: Rename output property `publicIpAddress` to `publicIPAddress` to keep consistent with the name in API
- [BREAKING CHANGE] `az network lb`: Update output property names in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid` profiles to keep consist with the `latest` profile.
- [BREAKING CHANGE] `az network cross-region-lb`: Remove in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid` profiles.
- [BREAKING CHANGE] `az network nic ip-config`: Rename output property `privateIpAddress` to `privateIPAddress` to keep consistent with the name in API
- [BREAKING CHANGE] `az network nic ip-config`: Rename output property `privateIpAllocationMethod` to `privateIPAllocationMethod` to keep consistent with the name in API
- [BREAKING CHANGE] `az network local-gateway`: Update output property names in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid` profiles to keep consist with the `latest` profile.
- [BREAKING CHANGE] `az network vpn-connection`: Update output property names in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid` profiles to keep consist with the `latest` profile.
- [BREAKING CHANGE] `az network vnet-gateway`: Update output property names in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid`

`hybrid` profiles to keep consist with the `latest` profile.

- [BREAKING CHANGE] `az network nic`: Update output property names in `2017-03-09-profile`, `2018-03-01-hybrid`, `2019-03-01-hybrid` and `2019-03-01-hybrid` profiles to keep consist with the `latest` profile
- [BREAKING CHANGE] `az network watcher flow-log`: Remove the deprecated command `configure`
- [BREAKING CHANGE] `az network vrouter`: Deprecate `vrouter` and use `routesserver` instead
- [BREAKING CHANGE] `az network watcher connection-monitor endpoint add`: Remove deprecated parameters `filter-item` and `filter-type`
- `az network nsg rule list`: Fix `-o table` cannot be used
- `az network private-endpoint-connection`: Add provider
`Microsoft.Monitor/Accounts`
- `az network express-route gateway connection create/update`: Add parameters `--inbound-route-map` and `--outbound-route-map` to support route map
- Fix #25408: `az network application-gateway rule create`: Creation fails with `--redirect-config` when there are multiple pools
- `az network private-endpoint-connection`: Add provider
`Microsoft.DBforMySQL/flexibleServers`

Packaging

- Remove `openssl1.1-compat` and use `openssl-dev` in docker image
- Support ARM64 on Linux

PolicyInsights

- Fix #25538: `az policy remediation create`: Fix the `Required` property `'policyAssignmentId'` not found in JSON error

RDBMS

- `az postgres flexible-server migration`: Use Cloud supplied URL's rather than hardcoded management URL's
- `az mysql flexible-server replica create`: Add `--location` to support specifying replica location
- `az mysql flexible-server update`: Fix `--storage-auto-grow` parameter unable to be set

Role

- `az role assignment create`: Show warning if `--scope` argument is not specified: `--scope` argument will become required for creating a role assignment in the breaking change release of the fall of 2023. Please explicitly specify `--scope`.
- Migrate `azure-mgmt-authorization` SDK to Track 2 and bump API version to 2022-04-01

Service Bus

- `az servicebus namespace`: Add `--premium-messaging-partitions` to support ServiceBus Namespace ScaleSet

Service Connector

- `az connection create`: Add new param `--customized-keys`

SQL

- `az sql instance-failover-group`: Add `--secondary-type` parameter to create and update commands
- `az sql mi db restore`: Add tags parameter
- `az sql mi create/update`: Add `--zone-redundant` to support zone redundancy
- `az sql db tde-key revalidate/revert`: New commands to revert and revalidate the TDE protector key for the database and SQL server
- `az sql db create/update/show`: Add `--keys`, `--encryption-protector`, `--assign-identity`, `--user-assigned-identity-id` parameters to support Per DB CMK

SQL VM

- `az sql vm create/update`: Deprecate `--sql-mgmt-type` argument
- `az sql vm update`: Update no longer requires the mode to be sent as full
- `az sql vm enable-azure-ad-auth/validate-azure-ad-auth`: New commands for Sqlvm Azure AD authentication

Storage

- `az storage blob copy start-batch`: Add `--destination-blob-type` and `--tier`

- Fix #25402: `az storage account network-rule`: Support adding and removing multiple IPs

February 07, 2023

Version 2.45.0

ACR

- [BREAKING CHANGE] `acr manifest list-referrers`: Support OCI reference types and remove ORAS artifact reference types
- `az acr check-name`: Make command work with different profile

AKS

- Make ContainerInsights DataCollectionRuleName consistent with Portal and other onboarding clients
- `az aks upgrade`: Show warning if the Kubernetes version isn't supplied
- `az aks create`: Deprecate parameters `--aad-client-app-id`, `--aad-server-app-id` and `--aad-server-app-secret`
- `az aks update-credentials`: Deprecate parameters `--reset-aad`, `--aad-client-app-id`, `--aad-server-app-id` and `--aad-server-app-secret`

App Service

- `az webapp create-remote-connection`: Update the host address of SSH tunnel from 0.0.0.0 to 127.0.0.1
- Add support to create ASPs with Memory Optimized Workers
- Fix #17720: `az functionapp function`: Add new command to list functions in a function app
- Fix #24285: `az webapp config access-restriction add`: Fix the bug that it does not support more than one Front Door ID in X-Azure-FDID
- Fix #23603: `az functionapp config set`: Add new parameter to set PowerShell version
- `az webapp config appsettings`: Register settings as deployment slot setting anytime when using `--slot-settings`
- `az webapp config backup delete`: Add new command to delete a backup of the webapp

ARM

- `az bicep`: Add configuration `bicep.use_binary_from_path`. Possible values include `if_running_in_ci` (default) and Booleans
- `az bicep`: Add configuration `bicep.check_version` that accepts Boolean values. If set to `False`, version checks for Bicep CLI will be disabled
- `az deployment what-if`: Fix an issue where formatting nested array changes throws an exception
- Fix #25022: `az resource tag`: Fix the issue of `the serializedData field is missing or null` when updating tag for Microsoft.insights/workbooks

ARO

- `az aro create/update`: Add NetworkContributor role to NAT Gateways in Cluster Resource Group when creating or updating clusters
- `az aro create`: Change `--pull-secret` parameter to no longer require `@` prefix on filenames

Backup

- `az backup vault`: Add new parameter `--public-network-access` to support enabling public network access for the backup vault
- `az backup vault create`: Add new parameter `--immutability-state` to support configuring immutability settings for the backup vault

Batch

- Fix #24007: `az batch pool create`: Fix bug that caused 'MissingRequiredProperty' error when parameter `--encryption-key-identifier` is used

Compute

- `az image builder identity assign`: Add this command to add managed identity to an existing image builder template
- `az image builder identity remove`: Add this command to remove managed identity from an existing image builder template
- `az image builder identity show`: Add this command to display managed identity of an existing image builder template
- `az vmss reimagine`: Let `--instance-id` support multiple ids

- Fix #25308: `az disk create`: Fix help message for creating a standard disk for uploading blobs
- `az vmss create/update`: Add new parameter `--enable-osimage-notification` to support enabling OS image scheduled event
- `az vmss create`: Add new parameter `--max-surge` to support enabling rolling upgrade policy max surge

Cosmos DB

- `az managed-cassandra datacenter update`: Add support to update `--sku`

DMS

- `az dms project task create`: Update DMS MySQL API to support new migration types

Feedback

- Stop including error messages in the feedback body

IoT

- `az iot hub wait`: Add wait commands
- `az iot hub delete`: Fix functionality issue for parameter `--no-wait`

Key Vault

- `az keyvault security-domain restore-blob`: Support restoring blob offline
- `az keyvault security-domain upload`: Add `--restore-blob` to prevent exposing keys in online environment

NetAppFiles

- `az netappfiles volume update`: Fix volume patch dataprotection props

Network

- [BREAKING CHANGE] `az network application-gateway ssl-profile`: Rename output property `verifyClientCertIssuerDn` to `verifyClientCertIssuerDN` to keep

consistent with the name in API

- [BREAKING CHANGE] az network cross-region-lb frontended-ip: Rename output properties with `publicIp` prefix to `publicIP` prefix to keep consistent with the names in API response
- [BREAKING CHANGE] az network lb frontended-ip: Rename output properties with `publicIp` prefix to `publicIP` prefix to keep consistent with the names in API response
- [BREAKING CHANGE] az network lb frontended-ip: Rename output properties with `privateIp` prefix to `privateIP` prefix to keep consistent with the names in API response
- [BREAKING CHANGE] az network lb inbound-nat-pool: Rename output property `enableFloatingIp` to `enableFloatingIP` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb inbound-nat-pool: Rename output property `frontendIpConfiguration` to `frontendIPConfiguration` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb inbound-nat-rule: Rename output property `enableFloatingIp` to `enableFloatingIP` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb inbound-nat-rule: Rename output property `frontendIpConfiguration` to `frontendIPConfiguration` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb rule: Rename output property `enableFloatingIp` to `enableFloatingIP` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb rule: Rename output property `frontendIpConfiguration` to `frontendIPConfiguration` to keep consistent with the name in API
- [BREAKING CHANGE] az network cross-region-lb rule: Rename output property `enableFloatingIp` to `enableFloatingIP` to keep consistent with the name in API
- [BREAKING CHANGE] az network cross-region-lb rule: Rename output property `frontendIpConfiguration` to `frontendIPConfiguration` to keep consistent with the name in API
- [BREAKING CHANGE] az network lb outbound-rule: Rename output property `frontendIpConfigurations` to `frontendIPConfigurations` to keep consistent with the name in API
- [BREAKING CHANGE] az network cross-region-lb address-pool: Rename output property `loadBalancerFrontendIpConfiguration` to `loadBalancerFrontendIPConfiguration` to keep consistent with the name in API
- [BREAKING CHANGE] az network cross-region-lb address-pool create: Replace preview argument `--backend-addresses-config-file --config-file` by `--backend-`

`addresses` which supports Json, files and shorthand syntax formats

- [BREAKING CHANGE] `az network bastion`: Move Azure Bastion to Azure CLI Extension `bastion`
- Fix #25130: `az network list-usages`: `-o table` cannot be used
- Fix #25124: `az network vnet-gateway create`: Active-Active gateway fails with insufficient IP configurations
- `az network dns zone export`: Fix the export to emit all ALIAS records for a particular record set name
- `az network public-ip create`: Add parameter `--ddos-protection-plan` to link a DDoS protection plan to public IP
- Fix #25181: `az network nsg rule create`: Use `*` as default value for protocol
- `az network cross-region-lb address-pool update`: Add new command to update an address-pool

PolicyInsights

- `az policy attestation`: Add new command groups to manage resource policy attestation

RDBMS

- `az mysql flexible-server create/update`: Add `--auto-scale-iops` to enable or disable autoscale of iops
- `az mysql flexible-server start/stop`: Add no-wait support
- `az postgres flexible-server start/stop`: Add no-wait support
- `az postgres flexible-server migration`: Change behavior of cancel/cutover and added Offline Flag for FMS based migrations

Service Bus

- `az servicebus topic subscription rule create/update`: Add `--correlation-filter` to support custom filters

SQL

- `az sql midb recover`: Add support for managed database recover creation option
- `az sql recoverable-midb show`: Add support for getting geo replicated backup
- `az sql recoverable-midb list`: Add support for listing geo replicated backups

- `az sql db geo-backup restore/show/list`: New commands to manage geo redundant backups
- `az sql db threat-policy`: Change expiration version for cmd group to 2.49.0
- `az sql mi dtc`: Add managed instance DTC commands
- `az sql midb restore`: Add support for cross-subscription restore
- `az sql db geo/ltr-backup restore`: Add service objective parameter to ltr restore and geo restore

Storage

- `az storage account create`: Ongoing breaking change warning for disallowing blob public access by default
- `az storage container immutability-policy create`: Allow user to not specify `--resource-group`

January 11, 2023

Version 2.44.1

Network

- Hotfix: Fix #25086: `az network lb probe`: Expose parameter `--probes`

January 10, 2023

Version 2.44.0

ACR

- `az acr manifest`: Support oci artifact manifest

AKS

- `az aks create`: Add new parameter `--data-collection-settings` to support for AKS Monitoring Addon in MSI auth mode
- `az aks install-cli`: Automatically add the installation directories to system path on windows
- `az aks create/update`: Add support for KEDA workload auto-scaler

APIM

- `az apim api create`: Parse URL to detect Protocol and API type

App Service

- Fix #23488: `az appservice plan create`: Fix zone redundant ASP creation fails for ASEv3
- Fix #24858: Support for new isolated v2 (I4v2, I5v2, I6v2) SKUs
- `az appservice ase upgrade/send-test-notification`: Add new commands for ASE to support ASE upgrade and sending test notifications
- `az appservice ase update`: Add `--allow-incoming-ftp-connections` to allow incoming FTP connections
- `az appservice ase update`: Add `--allow-remote-debugging` to allow remote debugging
- Fix #19893: `az appservice plan create`: Fix the bug that cannot create app plan in a different subscription for ASEv3
- Fix #16478: `az functionapp cors credentials`: Add enable/disable CORS creds commands
- Fix #22934: `az functionapp delete`: Add new parameter `--keep-empty-plan` to support keeping empty app service plan
- Fix #19469: `az functionapp vnet-integration add`: Add consumption plan validation
- `az staticwebapp functions link`: Add new parameter `--environment-name` to support setting the environment name of static site

ARM

- Fix #24810: Support ARM64 architecture for Bicep installation

Batch

- `az batch pool create/set`: Add new parameter `--target-communication` to support setting the desired node communication mode for the pool

Compute

- Fix #24896: `az vm create`: Fix the bug that the VM cannot be created from ACG image to other resource group

- `az disk create`: Add new parameter `--performance-plus` to support boosting the performance target
- `az vm list`: Add new parameter `--vmss` to support querying VM instances in a specific VMSS
- `az sig image-version create/update`: Add parameters `--target-edge-zone-encryption` and `--target-edge-zones` to support edge zones

Container

- `az container export`: Fix export when identity is set

Key Vault

- `az keyvault key create`: Support OKP key and Ed25519 curve

Monitor

- [BREAKING CHANGE] `az monitor action-group test-notifications create`: Remove notification in resource group and subscription level
- `az monitor diagnostic-settings create`: Create with resource id and export without workspace

NetAppFiles

- `az volumes qouta-rule create`: Add volume quota rule create command
- `az volumes qouta-rule show`: Add volume quota rule show command
- `az volumes qouta-rule list`: Add volume quota rule list command
- `az volumes qouta-rule update`: Add volume quota rule update command
- `az volumes qouta-rule delete`: Add volume quota rule delete command

Network

- `az network vnet-gateway create`: Add parameter `--edge-zone-vnet-id` for local gateway
- Fix #24853: `az network nsg rule create`: `--destination-asgs` and `--source-asgs` cannot be used
- Fix #24883: `az network application-gateway stop/start`: Add missed parameter `--ids`
- `az network watcher packet-capture create`: Resolve local path issue for Linux VM

- `az network lb update`: Expose parameter `--tags`

Redis

- `az redis import/export`: Add new optional parameter `--preferred-data-archive-method`
- `az redis server-link`: Linked server has two new properties:
`geoReplicatedPrimaryHostName` and `primaryHostName`

Security

- `az security alert update`: `--status` now support `resolve` and `inprogress`

Service Connector

- `az connection`: Support local connection which allows local environment to connect Azure resource
- Fix #24806: `az webapp connection create mysql-flexible`: Fix mysql connection command with `--system-identity`

SQL

- `az sql server/db/mi/midb advanced-threat-protection-setting show/update`: Support `advanced-threat-protection-setting` commands
- `az sql db threat-policy`: Declare deprecation of this command group in version 2.45.0
- `az sql db`: Add `--preferred-enclave-type` argument

Storage

- `az storage blob copy start`: Fix `--tier` to support setting blob tier when copying

December 06, 2022

Version 2.43.0

Core

- `aaz`: Fix `has_value` function for list, dict and object arg types (#24625)

- `aaz`: Support argument preview and experimental (#24637)
- `aaz`: Add registered property for `AAZBaseArg` (#24640)
- `aaz`: Add `aazlist` and `aazdict` args transform help functions (#24690)
- `aaz`: Support camel case key in `AAZObjectType` (#24771)
- Disable removing and updating system extension in Cloud Shell (#24361)

AKS

- `az aks enable-addons`: Add `--enable-syslog` parameter to monitoring addon
- `az aks nodepool`: Unify the option names used to specify the nodepool name and cluster name. For nodepool name, option names are `--nodepool-name`, `--name` and `-n`. For cluster name, option name is `--cluster-name`
- `az aks nodepool add`: Support the new SKU Mariner for parameter `--os-sku`

App Config

- `az appconfig`: Update raised errors in app config command module

App Service

- `az staticwebapp backends link`: Link an backend to a static webapp. Also known as "Bring your own Backend."
- `az staticwebapp backends unlink`: Unlink backend from a static webapp
- `az staticwebapp backends show`: Show details on the backend linked to a static webapp
- `az staticwebapp backends validate`: Validate an backend for a static webapp
- `az webapp config snapshot restore`: Fix the `AttributeError str object has no attribute get`
- `az appservice plan create/update`: Add new environment SKU for parameter `--sku`
- `az staticwebapp create`: Add new parameter `--login-with-ado` to create azure dev ops token automatically
- Fix #24506: `az functionapp keys set/delete`: Update the wrong accepted parameter value `systemKey` to `systemKeys` for `--key-type`
- `az webapp create`: Add `--public-network-access` parameter to support enabling public access
- `az staticwebapp hostname show`: Fix dns-txt-token validation command to show command

- Fix #24620: `az webapp create`: Improve the error message to show that the `az webapp list-runtimes` command depends on the specified runtime

ARM

- `az deployment mg create`: Add new parameter `--mode` to support setting the mode for deploying resources
- `az group lock list`: Mark the `--resource-group` as required in help message
- `az bicep install`: Address issue installing bicep on non-musl default systems with musl

Backup

- `az backup restore restore-disks`: Allow `--disk-encryption-set-id` for cross region restore

Compute

- Fix #24624: `az sig image-version create`: Fix the error that the `--os-vhd-storage-account` must be a managed disk or snapshot

IoT

- Fix #22257: `az iot dps linked-hub create`: Improve error handling for linked hubs
- `az iot hub create/delete`: Add `--no-wait` parameter to support no wait operation

Key Vault

- `az keyvault`: Add check-name command, support Security Domain Properties

Monitor

- `az monitor diagnostic-settings`: Add `--marketplace-partner-id` parameter

Network

- `az network bastion rdp`: Allow rdp session customization
- `az network private-endpoint-connection`: Enable private link support for provider `Microsoft.DesktopVirtualization/hostpools` and

```
Microsoft.DesktopVirtualization/workspaces
```

- `az network application-gateway`: Support OCSP revocation check on client certificate
- `az network traffic-manager endpoint`: Add `--always-serve` to manage the health check on endpoints
- `az network public-ip create`: Fix `--ip-tags` cannot be used
- `az network private-endpoint-connection`: Add Provider

```
Microsoft.MachineLearningServices/registries
```

RDBMS

- `az postgres flexible-server geo-restore/replica`: Introduce read replicas and geo-restore
- `az postgres flexible-server upgrade`: Add major version upgrade for PostgreSQL flexible server
- `az postgres flexible-server create/update/restore/replica`: Postgres flex byok
- `az postgres flexible-server identity`: Add user managed identity operations for PostgreSQL flexible server
- `az postgres flexible-server create/update/ad-admin`: Add Azure Active Directory Administrator operations for PostgreSQL flexible server

Service Connector

- `az webapp/spring/containerapp connection create mysql`: Deprecate mysql single server connection command

SQL

- `az sql server ipv6-firewall-rule`: Add new command group for AZ SQL server IPv6 firewall rule

SQL VM

- `az sql vm update`: Deprecate the `--yes` prompt to upgrade SqlIaaSAgent extension to full mode
- `az sql vm create/update`: Add `--least-privilege-mode` to take minimal permissions on their SQL Server
- `az sql vm group create/update`: Add `--cluster-subnet-type` to support High Availability configuration

Storage

- Fix #23893, #24528: `az storage account show-connection-string/keys renew`: Fix resource group auto completion
- Fix #23216: `az storage file upload-batch`: Fix `--dryrun` to show correct file paths
- `az storage blob copy start`: Add `--destination-blob-type` to allow switching between blob types when copying
- `az storage account encryption-scope list`: Add `--filter`, `--include`, `--maxpagesize` to support advanced list
- `az storage account failover`: Add `--failover-type` to support planned failover

November 01, 2022

Version 2.42.0

ACR

- `az acr task update`: Fix logic issue for updating encoded task

AKS

- Fix #24188: `az aks list`: Fix pagination handling error
`ContainerServiceClientConfiguration object has no attribute api_version` when there are many list results
- Fix #24188: `az aks nodepool list`: Fix pagination handling error
`ContainerServiceClientConfiguration object has no attribute api_version` when there are many list results
- `az aks create/update`: Add new parameters `--enable-blob-driver` and `--disable-blob-driver` to enable/disable Blob CSI Driver
- `az aks create/update`: Add new parameter `--enable-oidc-issuer` to support enabling oidc issuer feature
- `az aks oidc-issuer rotate-signing-keys`: Add new command to support rotating oidc issuer service account signing keys

APIM

- `az apim create/update`: Add `--public-network-access` to support specifying whether or not public endpoint access is allowed for this API management service

- `az apim create/update`: Add `--disable-gateway` to support disabling gateway in the master region

App Config

- `az appconfig`: Update raised errors in app config command module

App Service

- Fix #23050: `az functionapp deployment source config-zip`: Fix the bug that zip deployment will fail if app settings contain any values of null

Backup

- `az backup restore restore-disks`: Update Cross Zonal Restore behaviour for ZRS vaults and primary region CRR scenarios
- `az backup job show`: Change subtask start/end time from minimum value to null for ongoing or yet-to-start operation

Compute

- `az vm run-command create/update`: Change help messages and add examples for `-output-blob-uri` parameter to illustrate that `--output-blob-uri` must be SAS URI
- Fix #24187: `az vm list`: Fix the AttributeError 'ComputeManagementClientConfiguration' object has no attribute 'api_version'
- `az vm extension list`: Add new parameter `--ids` to support listing extensions by VM id
- `az sig image-version create/update`: Add `--allow-replicated-location-deletion` to support removing gallery image version from replicated regions
- Fix #24263: `az snapshot create`: Fix the KeyError 'IMPORT_ENUM' when creating snapshot from source blob uri
- `az sig image-version update`: Support `excludeFromLatest` for `--add` parameter to exclude this image version when using the latest version of image definition
- `az sig image-version update`: Support `safetyProfile.allowDeletionOfReplicatedLocations` for `--set` parameter to allow users to remove the gallery image version from replicated regions

HDInsight

- [BREAKING CHANGE] `az hdinsight create`: Remove the enum value 1.0 and 1.1 from the `--minimal-tls-version`, HDInsight doesn't support TLS version which is less than 1.2 now.

IoT

- `az iot hub create`: Enforce data residency property on hubs created in `qatarcentral`

NetAppFiles

- `az netappfiles account renew-credentials`: Add `renew-credentials` command to renew identity credentials that are used to authenticate to key vault, for customer-managed key encryption

Network

- `az network public-ip`: Add alias `--ddos-protection-mode` to `--protection-mode`
- `az network custom-ip prefix`: Add parameters `--asn`, `--geo`, `--no-internet-advertise` and so on
- Fix #21551: `az network nic ip-config update`: ASGs update with multiple IP configurations
- Fix #24169: `az network application-gateway waf-policy managed-rule exclusion rule-set remove`: Remove exclusion with different matchers
- Fix #24377: `az network public-ip create`: Derive Public IPs in different resource group from Public IP Prefix
- `az network lb probe`: Support probe threshold via `--probe-threshold`

RDBMS

- [BREAKING CHANGE] `az postgres flexible-server migration show`: Remove `--level` parameter
- [BREAKING CHANGE] `az postgres flexible-server migration delete`: Remove this command. Deleting a migration is not supported for now.
- [BREAKING CHANGE] Change `az postgres flexible-server migration update --cutover` to `az postgres flexible-server migration update --cutover db1 db2 db3`
- `az postgres flexible-server migration create`: Add `--migration-mode` to support offline and online(with CDC) migrations. Default mode when `--migration-mode` not passed will be offline.

- Add `az postgres flexible-server migration update --cancel db1 db2 db3` to cancel a migration.

Resource

- `az resource delete`: Add new parameter `--no-wait` to support not waiting the long-running operation to finish

Role

- `az role assignment create`: Support bring-your-own role assignment name
- `az role assignment delete`: If `--ids` is provided, ignore other arguments, instead of raising error

SQL

- `az sql midb log-replay start`: Add `--storage-identity` parameter

Storage

- `az storage account show-connection-string/keys renew`: Update options for `--key` parameter
- `az storage account create/update`: GA `--key-vault-federated-client-id`

Synapse

- `az synapse workspace create`: Add parameter `--managed-resource-group-name`
- `az synapse spark pool`: Add parameter `--enable-dynamic-executor-allocation`

October 11, 2022

Version 2.41.0

Core

- Support Continuous Access Evaluation
- PREVIEW: Support Web Account Manager (WAM) login on Windows. To opt in, run
`az config set core.allow_broker=true`
- Revert #23514: Rename entry script `az.ps1` to `azps.ps1`

- `aaz`: Support `yaml` file as value for compound arguments
- `aaz`: Fix recursion depth exceeded for `to_serialized_data` of AAZObject
- `aaz`: Support customized life cycle callback functions for `AAZCommand`
- `aaz`: Fix two AAZObjects or AAZDicts comparison

ACS

- [BREAKING CHANGE] `az acs`: Remove the deprecated command group

AD

- Support special characters in user principal name

AKS

- Fix #23779: `az aks install-cli`: Support determining the arch of binaries based on system information

API

- Fix #20863: `az apim api import`: Fix the issue to import GraphQL API's using graphqllink

App Config

- [BREAKING CHANGE] `az appconfig kv import`: Add validation to JSON file import to ensure that only valid JSON objects are imports
- [BREAKING CHANGE] `az appconfig kv export`: Update the array conversion logic to prevent dropping keys during export
- `az appconfig kv export`: Fix MemoryError while exporting large stores
- `az appconfig replica`: New command group to support geo-replication
- `az appconfig kv export`: Support exporting app configuration settings as references to App Service
- `az appconfig kv import`: Ensure app configuration references are not imported from App Service
- `az appconfig feature filter update`: Add new command to support updating functionality for feature filters

App Service

- `az functionapp deployment github-actions`: Add support for linux powershell runtimes
- `az functionapp deployment github-actions`: Fix issue where publish profile would not be populated before the github action was run
- `az webapp up`: No longer show status during linux deployments
- `az webapp deployment source config-zip`: No longer show status during linux deployments

ARM

- `az deployment group what-if`: Fix an issue where `complete` deployment mode does not work

Backup

- `az backup policy`: Add support for Smart Tiering policy

Compute

- [BREAKING CHANGE] `az vmss create`: Update NAT pool to NAT rule V2 for Standard LB SKU when creating VMSS
- `az vm/vmss create`: `--enable-secure-boot` is set to True by default when the `--security-type` used by the VM/VMSS creation is `TrustedLaunch`
- `az restore-point create`: Add new parameter `--consistency-mode` to support setting consistency mode
- `az vmss create/update`: Add new parameters `--priority-count` and `--priority-percentage` to support setting priority mix policy
- `az vm/vmss create/update`: Add new parameter `--disk-controller-type` to support setting disk controller type
- `az disk create`: Add warning log in three scenarios to later support creating disk with Gen2 and TLVM as default
- `az vmss create`: Add new parameter `--nat-rule-name` to specify the name of NAT rule V2 when creating a new load balancer (NAT rule V2 is used to replace NAT pool)

Cosmos DB

- `az cosmosdb mongodb role/user definition`: New command groups for enforcing RBAC on Cosmos DB Mongo accounts

- `az cosmosdb create/update`: GA mongo server version

Event Hubs

- [BREAKING CHANGE] `az eventhubs namespace update`: Remove `--key-source`, `--key-name`, `--key-vault-uri` and `--key-version`. Please use `az eventhubs namespace encryption` to manage keys
- [BREAKING CHANGE] `az eventhubs namespace create/update`: Remove `--identity`. Please use `--mi-user-assigned` and `--mi-system-assigned` parameters and `az eventhubs namespace identity` commands
- [BREAKING CHANGE] `az eventhubs namespace create/update`: Remove `--default-action` and `--enable-trusted-service-access`. Please use `az eventhubs namespace network-rule update` command instead

Key Vault

- [BREAKING CHANGE] `az keyvault create/update`: Finally remove `--enable-soft-delete` parameter
- Fix #23527: `az keyvault secret set`: Add alias `--content-type` for `--description`

Monitor

- [BREAKING CHANGE] `az monitor diagnostic-settings list`: Drop `value` property in output, return a list instead of a dict
- `az monitor autoscale`: Upgrade monitor autoscale api version
- `az monitor autoscale`: Add predictive metric show cmd

NetAppFiles

- `az netappfiles account create`: Add optional parameters `--key-name`, `--key-source`, `--keyvault-resource-id`, `--user-assigned-identity`
- `az netappfiles account update`: Add optional parameters `--key-name`, `--key-source`, `--keyvault-resource-id`, `--user-assigned-identity`
- `az netappfiles volume create`: Add optional parameters `--smb-access-based-enumeration`, `--smb-non-browsable`, `--delete-base-snapshot`
- `az netappfiles resource`: Add new command `query-region-info`

Network

- [BREAKING CHANGE] az network watcher connection-monitor create: Deprecate classic connection monitor creation
- [BREAKING CHANGE] az network application-gateway waf-policy managed-rule rule-set: Change parameter --rules to --rule and support multi-properties
- [BREAKING CHANGE] az network vnet: Deprecate parameter --defer
- [BREAKING CHANGE] az network public-ip: Change publicIpAllocationMethod to publicIPAllocationMethod
- [BREAKING CHANGE] az network public-ip: Change publicIp.publicIpPrefix to publicIp.publicIPPrefix
- [BREAKING CHANGE] az network public-ip: Change publicIpAddressVersion to publicIPAddressVersion
- Fix #23884: az network application-gateway rule create: Compatible with v1 SKU
- az network private-endpoint-connection: Add Provider Microsoft.AgFoodPlatform/farmBeats
- az network application-gateway waf-policy managed-rule rule-set: Support per rule actions in web application firewall
- az network public-ip: Support ddos protection mode via --protection-mode

Packaging

- Drop Mariner 1.0 RPM package

RDBMS

- az mysql flexible-server update: Expose --geo-redundant-backup argument
- az mysql/postgres flexible-server create/update: Deprecate Enabled for --high-availability argument
- az mysql flexible-server stop: Change stopped time logging message
- az mysql flexible-server ad-admin delete: Disable aad_auth_only when dropping AAD admin
- az mysql flexible-server identity remove: Allow removing all identities in a MySQL server

Reservations

- Move commands from azure-cli to reservation extension

Service Bus

- [BREAKING CHANGE] `az servicebus namespace create/update`: Remove `--default-action`. Please use `az servicebus namespace network-rule update` command instead
- `az servicebus queue/topic create/update`: Support setting max message size
- `az servicebus topic subscription create`: Support client affine

Service Connector

- `az spring-cloud connection create postgres`: Add `--system-identity` for springcloud-postgres connection

SQL

- `az sql server audit-policy show`: Add `isManagedIdentityInUse` info in output

Storage

- `az storage blob/container`: Support `--account-name` for non-standard account URL
- `az storage account update`: Fix ADProperties wipe out issue when updating `--default-share-permission`
- Fix #19311: `az storage remove`: Add support for connection-string

September 06, 2022

Version 2.40.0

ACR

- `az acr config authentication-as-arm show`: Add new command to support showing the configured 'Azure AD authenticate as ARM' policy
- `az acr config authentication-as-arm update`: Add new command to support updating 'Azure AD authenticate as ARM' policy
- `az acr config soft-delete show`: Add new command to show soft-delete policy
- `az acr config soft-delete update`: Add new command to update soft-delete policy
- `az acr repository list-deleted`: Add new command to list deleted repositories
- `az acr manifest list-deleted`: Add new command to list deleted manifests
- `az acr manifest list-deleted-tags`: Add new command to list deleted tags

- `az acr manifest restore`: Add new command to restore deleted manifests and tags
- `az acr network-rule`: Deprecate params `--subnet` and `--vnet-name`
- `acr config`: Fix bug in some commands that would in certain circumstances attempt to pull a nonexistent model from SDK

AKS

- Fix #23468: `az aks nodepool wait` crashes with error "'Namespace' object has no attribute 'nodepool_name'"
- `az aks check-acr`: Append acr suffix to option `--acr` according to cloud env
- `az aks`: Add `--gpu-instance-profile` for Nvidia multi-instan...
- `az aks update`: Update without args prompts to reconcile
- `az aks create/update`: Add new parameters `--enable-disk-driver` and `--disable-disk-driver` to enable/disable AzureDisk CSI Driver. When creating new cluster, AzureDisk CSI Driver is enabled by default.
- `az aks create/update`: Add new parameters `--enable-file-driver` and `--disable-file-driver` to enable/disable AzureFile CSI Driver. When creating new cluster, AzureFile CSI Driver is enabled by default.
- `az aks create/update`: Add new parameters `--enable-snapshot-controller` and `--disable-snapshot-controller` to enable/disable CSI Snapshot Controller. When creating new cluster, CSI Snapshot Controller is enabled by default.
- `az aks nodepool add`: Add option `Windows2019`, `Windows2022` to `--os-sku` parameter
- Fix #23653: `az aks create`: Fix the CrashLoopBackOff issue when set `--network-policy` to 'Calico'

App Service

- Fix #23417: `az functionapp github-actions add`: Fix the functionapp github actions on java
- `az functionapp list-runtimes`: Add linuxFxVersion to output
- `az webapp up`: Show status during deployment for linux apps
- `az webapp deployment source config-zip`: Show status during deployment for linux apps
- `az logicapp deployment`: Add a new command group to support managing logic app deployments
- `az logicapp scale`: Add a new command to support scaling a logic app

- `az logicapp config`: Add a new command group to support configuring a logic app
- `az logicapp update`: Add a new command to support updating a logic app

ARM

- `az bicep`: Use `AZURE_CLI_DISABLE_CONNECTION_VERIFICATION` when checking Bicep CLI versions

Backup

- `az backup vault create/backup-properties set`: Add support for Alert Settings
- Fix #23655: `az backup restore restore-disks`: Support storage account being in a different resource group

Batch

- Fix #23445: `az batch pool supported-images list`: Fix the `NoneType object has no attribute startswith` bug for getting supported images list

Compute

- `az vm run-command invoke`: Add new parameters `--no-wait` to support not waiting for the long running operation to finish
- Fix #23194: `sig image-version create`: Fix the `Parameter tags must be of type dict` error when `--tags` parameter is passed as `key=value` pairs
- Fix #23540: `az ppg create`: Fix the `Parameter tags must be of type dict` error when `--tags` parameter is passed as `key=value` pairs
- `az sig update`: Add parameters to support updating gallery from private to community
- `az sig share reset`: Update gallery from community to private
- `az vm/vmss create`: `--enable-vtpm` is set to `True` by default when the `--security-type` used by the VM/VMSS creation is `TrustedLaunch`
- Fix #23341: `az vm list-skus`: Fix filtering out VM sizes that are available regionally when they are restricted in all zones
- `az vm run-command show/list`: Add validation and refine help message for parameter combination
- `az identity federated-credential`: Add subgroup to support managing federated identity credentials of existing user assigned identities

Cosmos DB

- `az cosmos db service`: Add service support for cosmosDB
- `az cosmosdb gremlin graph`: Add analyticalStorageTTL property to sql containers

Feedback

- `az survey`: New command for CLI survey

Monitor

- `az monitor action-group test-notifications create`: Add new command
- `az monitor metric alert`: Support metric namespace with dash
- `az monitor action-group create`: Add optional parameter `--location`

NetAppFiles

- `az netappfiles volume create`: Add optional parameter `--kv-private-endpoint-id`
- `az netappfiles volume-group create`: Add optional parameter `--kv-private-endpoint-id`
- `az netappfiles volume update`: Add optional parameter `--cool-access`
- `az netappfiles volume update`: Add optional parameter `--coolness-period`
- `az netappfiles pool update`: Add optional parameter `--cool-access`

Network

- `az network application-gateway create`: Support rule priority field provided as part of configuration
- `az network private-endpoint-connection`: Add `Microsoft.OpenEnergyPlatform/energyServices` provider
- Fix #22594: `az network bastion create`: Add no wait support for bastion create
- Fix #23525: `az network bastion create/update`: Add missing arguments and update command
- `az network watcher packet-capture create`: Add VMSS support in packet capture

Packaging

- Build RPM for RHEL 9 and CentOS Stream 9

RDBMS

- `az mysql flexible-server upgrade`: Add major version upgrade for MySQL flexible server
- `az mysql/postgres flexible-server backup`: Add backup commands for flexible servers
- `az postgres flexible-server create/update`: Add `SameZone` for HA in PostgreSQL flexible server
- `az mysql flexible-server create/update/restore/geo-restore/replica`: Add BYOK for MySQL Flexible Server
- `az mysql flexible-server identity/ad-admin`: User Identity and AAD Admin for MySQL flexible server

Security

- `az security security-solutions-reference-data`: Add new command group
- `az security security-solutions`: Add new command group

Service Bus

- `az servicebus namespace create/update`: Support specifying `--min-tls`
- `az servicebus namespace network-rule update`: Support updating network rules for given namespace

Service Connector

- `az spring connection`: Update description after spring app renames

SignalR

- `az signalr custom-domain`: Support custom domain
- `az signalr custom-certificate`: Support custom certificate

SQL

- `az sql mi endpoint-cert`: New command group to manage endpoint certificates
- `az sql mi partner-cert`: New command group to manage partner certificates
- `az sql mi link`: New command group to manage instance link

Storage

- `az storage fs file set-expiry`: New command to support setting expiry for files in ADLS Gen2 file system
- `az storage account create/update`: Add `--enable-files-aadkerb` to support AAD Kerberos authentication for Azure Files
- `az storage account local-user`: New command group to manage identities when using SFTP
- `az storage account create/update`: Add `--enable-sftp` and `--enable-local-user` to support SSH File Transfer Protocol
- `az storage fs create`: Support encryption scope
- `az storage fs directory/fs generate-sas`: Support generating SAS token with specified encryption scope

August 02, 2022

Version 2.39.0

ACR

- [BREAKING CHANGE] Update manifest list-referrers to comply with RC1 ORAS spec
- `az acr update`: Update networkRuleSet.defaultAction to deny when `--public-network-enabled` is disabled
- Fix #23340: `az acr task credential add`: Fix crashes when given a password but no username

AD

- `az ad app federated-credential`: Federated identity credential GA

Advisor

- Fix #11070: `az advisor recommendation disable`: Fix NoneType error

AKS

- Fix snapshot not resolved according to the subscriptions field in the `--snapshot-id` option

- `az aks check-acr`: Bump canipull to v0.1.0 to add 5s wait to avoid attach race condition
- `az aks update`: Fix the issue of `NoneType` error when updating the config of keyvault secret provider
- Remove warning message when using "BYO vnet + system MSI"
- Fix the bug related to AKS Monitoring MSI auth when the location value with spaces
- Fix #2457: Clarify subnet id description to resource id
- `az aks create`: Add new parameter `--host-group-id` to support Azure dedicated host
- `az aks nodepool add`: Add new parameter `--host-group-id` to support Azure dedicated host
- `az aks create/update`: Add new parameters `--enable-azure-keyvault-kms`, `--azure-keyvault-kms-key-id`, `--azure-keyvault-kms-key-vault-network-access`, `--azure-keyvault-kms-key-vault-resource-id` and `--disable-azure-keyvault-kms` to support Key Management Service feature with Azure Key Vault
- `az aks create`: Add `--network-plugin=none` support for BYO CNI
- `az aks create/update`: Add parameter `--http-proxy-config` to support setting HTTP Proxy configuration

App Service

- Fix #23135: `az functionapp plan create`: Add validation for the valid value of `--number-of-workers` option
- `az functionapp/logicapp create`: Add new `--https-only` parameter
- `az functionapp/webapp create`: Allow vnet integration for basic and elastic premium SKUs
- `az webapp list-runtimes`: Add Java 17 Support
- `az webapp create`: Add Java 17 Support
- `az webapp up`: Add Java 17 Support
- `az functionapp deployment github-actions add`: Add command to create GitHub actions to deploy to a Function App
- `az functionapp deployment github-actions remove`: Add command to remove Function App GitHub actions
- `az webapp deployment github-actions`: Add validation to ensure app is Web App

ARM

- Fix #23246: Fix interchanged policy samples

Backup

- `az backup protection backup-now`: Fix bug for SQL/HANA backup retention

Batch

- `az batch account network-profile show`: Add show network profile command for batch account
- `az batch account network-profile set`: Add set network profile command for batch account
- `az batch account network-profile network-rule list`: Add rule list command for batch account network
- `az batch account network-profile network-rule add`: Add rule add command for batch account network
- `az batch account network-profile network-rule delete`: Add rule delete command for batch account network
- `az batch account create`: Add managed identity support with `--mi-user-assigned` parameter
- `az batch account identity assign`: Add command to add identity to existing batch accounts
- `az batch account identity remove`: Add remove identity for existing batch accounts
- `az batch account identity show`: Add show identity for batch accounts
- `az batch pool create`: Update help text for `--json-file` to point to json schema

Compute

- `az ppg create/update`: Add parameter `--intentvmsizes` to specify possible sizes of VM that can be created in the proximity placement group
- `az ppg create`: Add parameter `--zone` to support specifying availability zone where the ppg should be created
- Fix #22995: `az image-version create`: Unbind the usage of `--target-region-encryption` and `--target-region-cvm-encryption`
- Fix #22654: `az vm run-command create/update`: Parameter `--protected-parameters` does not achieve the desired effect
- `az vmss run-command create/update`: Parameter `--protected-parameters` does not achieve the desired effect
- `az vmss create`: Add new parameter `--os-disk-delete-option` to support configuring whether the VM OS disks of Flex VMSS will be deleted or detached

upon VM deletion

- `az vmss create`: Add new parameter `--data-disk-delete-option` to support configuring whether the VM data disks of Flex VMSS will be deleted or detached upon VM deletion
- `az image builder create`: Add parameter `--staging-resource-group` to support custom resource group naming
- `az image builder validator`: Add subgroup to manage validate information of template
- `az vm disk detach`: Add parameter `--force-detach` to support force-detaching managed data disks from a VM

Container

- `az container create`: Add environment variable interpolation in container group yaml

Event Grid

- Add commands for partner and event-subscription customer facing features

Eventhub

- `az eventhubs namespace`: Add `--minimum-tls-version`
- `az eventhubs cluster`: Add `--supports-scaling`

IoT

- Change certificate loading to encode to b64 strings by default

Key Vault

- `az keyvault security-domain upload`: Fix `password must be bytes-like` for `--passwords`

Monitor

- `az monitor autoscale rule create`: Suppress warning from antlr
- `az monitor metrics alert create/update`: Suppress warning from antlr

Network

- `az network vnet subnet list-available-ips`: Get list of available IPs for subnet
- `az network private-endpoint-connection`: Enable private link support for provider `Microsoft.KubernetesConfiguration/privateLinkScopes`
- `az network private-endpoint-connection`: Enable private link support for provider `Microsoft.Dashboard/grafana`
- `az network dns zone export`: Add support for ALIAS record
- `az network dns zone import`: Add support for ALIAS record
- `az network application-gateway waf-policy custom-rule match-condition add`: Add validation for WAF custom rule condition
- `az network watcher flow-log`: Add support for `--vnet`, `--subnet`, `--nic` as target ID
- `az network private-endpoint create`: Add an example for creating with ASGs

Packaging

- Drop CentOS 7 RPM package
- Drop Python 3.6 support
- Build RPM for Fedora
- Drop Ubuntu 21.10 Impish Indri DEB package

Profile

- `az account list`: Add `TenantId` column to table output

RDBMS

- `az mysql flexible-server server-logs`: Add server logs for MySQL Flexible Server

Service Connector

- `az spring connection create eventhubs`: Add new parameter `--client-type kafka-springBoot`
- `az webapp connection create`: Add `--config-connstr` to support webapp connection strings
- `az webapp connection create`: Use webapp name and resource group from config

SQL

- `az sql log-replay stop`: Drop DB only if it was created with LRS

Storage

- `az storage fs undelete-path`: Encode `--deleted-path-name` automatically
- Fix #23179: `az storage file upload/upload-batch`: Fix `--content-md5` for upload, ignore `--content-md5` for upload-batch
- `az storage file show`: Fix JSON error when content-md5 is not None
- `az storage blob/file update`: Fix `--content-md5` TypeError
- `az storage container policy create`: No longer use default value for start and expiry time
- `az storage blob upload`: Add back `--socket-timeout` which has been renamed by SDK
- Fix #23262: `az storage blob metadata`: Add `--lease-id` back
- `az storage blob download/download-batch`: Add `--overwrite`

Synapse

- `az synapse workspace`: Add `--last-commit-id` for git repo config
- `az synapse ad-only-auth`: New command group for supporting synapse azure ad only authentication

June 21, 2024

Version 2.38.2

This version is only released for CentOS 7 and RHEL 7. CVE-2023-36052 is fixed in this version.

App Service

- [BREAKING CHANGE] Redact appsettings output on set/delete commands
- [BREAKING CHANGE] `az webapp config`: Redact webapp config connection-string and storage-account values
- [BREAKING CHANGE] `az functionapp`: Redact key value output on keys set
- [BREAKING CHANGE] `az functionapp/webapp deployment source update-token`: Redact tokens output on deployment source update-token
- [BREAKING CHANGE] Redact output on deployment source config

September 01, 2023

Version 2.38.1

This version is only available on CentOS 7 and RHEL 7.

App Service

- `az webapp ssh`: Backport #25141 to 2.38

July 05, 2022

Version 2.38.0

ACR

- `az acr`: Show replication region endpoint status in table output
- `az acr task run`: Add Dockerfile to source upload if context is local directory

AD

- `az ad app/sp update`: Support generic update `--set` on root level
- Support special characters in object names
- `az ad app federated-credential`: Support federated identity credentials

AKS

- `az aks get-credentials`: Fix permission prompt when saving config file to symlink
- `az aks command invoke`: Add support for `--no-wait`
- `az aks get-credentials`: Fix the command error when KUBECONFIG is empty
- `az aks nodepool stop/start`: Add nodepool stop/start bindings

APIM

- `az apim`: Update experimental flag to get out of experimental state
- `az apim deletedservice`: Add command group to support managing soft-deleted azure API Management services

App Config

- `az appconfig`: GA features - soft-delete, feature-filter, strict-import and disable-local-auth

App Service

- [BREAKING CHANGE] `az webapp up`: Remove premium container SKUs (PC2, PC3, PC4)
- [BREAKING CHANGE] `az appservice plan create/update`: Remove premium container SKUs (PC2, PC3, PC4)
- [BREAKING CHANGE] `az functionapp plan create`: Remove premium container SKUs (PC2, PC3, PC4)
- Fix #22722: `az webapp config ssl import` fixes to support new GraphAPI for SP queries
- `az webapp up`: Fix bug where runtime is detected even when `--runtime` is provided
- `az staticwebapp enterprise-edge`: Move command group from extension to official CLI
- `az appservice plan create`: Allow creating Hyper-V App Service Plans hosted on App Service Environments
- `az webapp/functionapp deployment slot create`: Allow using `--configuration-source` for apps with storage accounts added
- `az webapp up`: Fix bug when deploying to an App Service Environment (ASE) where the ASE is incorrectly categorized as an internal load balancing (ILB) ASE and fails validation
- Fix #20901: `az functionapp update`: Update `--slot` logic to work correctly

ARM

- Fix #22621: `az bicep build`: `--stdout` does not work
- Fix #22930: `az bicep generate-params`: Add support for bicep generate-params command
- `az deployment`: Fix the error message of ARM deployment to the correct JSON format

Backup

- `az backup restore restore-disks`: Add Cross Subscription Restore for IaaSVM ALR
- `az backup protection enable-for-vm`: Add a linux specific example
- `az backup protectable-item list`: SLAG container fetch failure bug fix

Bot Service

- [BREAKING CHANGE] `az bot create`: Remove `--kind`, `--password`, `--lang` arguments. Add `--app-type`, `--tenant-id`, `--msi-resource-id` arguments

Cognitive Services

- `az cognitiveservices account deployment create`: Support standard scale type

Compute

- `az disk create`: Fix the issue that specifying encryption type as `EncryptionAtRestWithPlatformKey` does not take effect when creating a disk
- `az disk update`: Fix the `(InvalidParameter) Resource xxx encrypted with platform key has disk encryption set id specified` error when updating the encryption type to platform managed keys
- `az sig image-version create`: Add new parameters `--virtual-machine` and `--image-version` to support creating image version from different source
- `az vm`: Support a new disk storage SKU Premiumv2_LRS
- `az sig show-community`: Add new command to support listing image versions in community gallery
- `az sig image-definition show-community`: Add new command to support getting an image in a gallery community
- `az sig image-definition list-community`: Add new command to support listing VM Image definitions in a gallery community
- `az sig image-version show-community`: Add new command to support getting an image version in a gallery community
- `az sig image-version list-community`: Add new command to support listing VM image versions in a gallery community
- `az sig share enable-community`: Add new command to support sharing gallery to community
- `az sig gallery-application version`: Add new parameter `--package-file-name` to specify the downloaded package file on the VM
- `az sig gallery-application version`: Add new parameter `--config-file-name` to specify the downloaded config file on the VM
- `az disk create`: Add support for `--gallery-image-reference` to allow creating disk from shared gallery image version or community gallery image version
- `az disk create`: Add support for `--source` to allow creating a disk from disk restore point

- `az vm/vmss application set`: Add new parameter `--treat-deployment-as-failure` to treat any failure in the gallery application version as deployment failure
- `az vm image list`: Add parameter `--architecture` to filter image with its architecture
- `az disk-encryption-set create`: The `--encryption-type` parameter supports new value `ConfidentialVmEncryptedWithCustomerKey` for confidential VM
- `az disk create`: The `--security-type` parameter supports new value `ConfidentialVM_DiskEncryptedWithCustomerKey` for confidential VM
- `az disk create`: Add new parameter `--secure-vm-disk-encryption-set` to provide ID or name of disk encryption set created with `ConfidentialVmEncryptedWithCustomerKey` encryption type
- `az disk-encryption-set create/update`: Add new parameter `--federated-client-id` to access key vault in a different tenant
- `az disk-encryption-set create`: Add new parameters `--mi-system-assigned` and `--mi-user-assigned` to support assigning system and user assigned identities during disk encryption set creation
- `az disk-encryption-set identity`: Add new command groups with parameters `--system-assigned` and `--user-assigned` to support managing system and user assigned identities on existing disk encryption set
- `sig list-community`: Add new command to support listing community gallery
- `sig list-community`: GA shared/community image gallery related feature
- `az vm/vmss create`: The `--security-type` parameter supports new value `ConfidentialVM` for Confidential VM
- `az vm/vmss create`: Add new parameter `--os-disk-security-encryption-type` to support setting the encryption type of the OS managed disk for Confidential VM
- `az vm/vmss create`: Add new parameter `--os-disk-secure-vm-disk-encryption-set` to allows users to provide ID or name for disk encryption set created with `ConfidentialVmEncryptedWithCustomerKey` encryption type
- `az disk create`: Add new parameter `--security-data-uri` to specify the blob URI of VHD to be imported into VM guest state
- `az disk create`: Add new parameter `--upload-type` to extend and replace `--for-upload` which supports standard disk only upload and OS Disk upload along with VM guest state
- `az disk grant-access`: Add new parameter `--secure-vm-guest-state-sas` to support getting security data access SAS on managed disk with VM guest state

Cosmos DB

- `az cosmosdb sql container create`: Add support to create containers with client encryption policy

Event Hubs

- `az eventhubs namespace application-group`: New command group to support management operations on EventHubs application groups
- `az eventhubs namespace network-rule update`: New command to update Network Rule Sets

IoT

- `az iot hub/dps certificate list`: Add table transform to certificate list commands

Key Vault

- `az keyvault role assignment`: Fix 'dict' object has no attribute 'object_id' error
- Fix #16390: `az keyvault set-policy`: Allow clearing permissions

Monitor

- `az monitor log-analytics query-pack`: Add query pack commands.
- `az monitor log-analytics update`: Support empty string for `--key-version`

NetAppFiles

- `az netappfiles account create`: Change `--location` to an optional parameter
- `az netappfiles pool create`: Change `--location` to an optional parameter
- `az netappfiles volume create`: Change `--location` to an optional parameter
- `az netappfiles snapshot create`: Change `--location` to an optional parameter
- `az netappfiles snapshot policy create`: Change `--location` to an optional parameter
- `az netappfiles snapshot policy update`: Change `--location` to an optional parameter
- `az netappfiles backup create`: Change `--location` to an optional parameter
- `az netappfiles backup-policy create`: Change `--location` to an optional parameter

- `az netappfiles volume-group create --help`: Fix typo in option global-placement-rules
- `az netappfiles volume create`: Add optional parameter `--zones`
- `az netappfiles volume replication list`: Add operation to list volume replications
- `az netappfiles volume reset-cifs-pw`: Add operation to reset CIFS password
- `az netappfiles volume relocate`: Add operation to relocate volume to a new stamp
- `az netappfiles volume finalize-relocation`: Add operation to finalize volume relocation
- `az netappfiles volume revert-relocation`: Add operation to revert volume relocation

Network

- [BREAKING CHANGE] `az network vnet subnet create`: Disable `PrivateEndpointNetworkPolicies` by default
- `az network application-gateway ssl-policy`: Support new SSL policy `CustomV2`
- `az network private-endpoint-connection`: Enable Private link support for provider `Microsoft.Authorization/resourceManagementPrivateLinks`
- Fix #22097: `az network dns zone import`: Fix importing zone files starting with space
- `az network public-ip prefix create`: Support cross-subscription association for Custom IP Prefix
- `az network public-ip create`: Reuse prefix info when creating Public IP

Packaging

- Use Python 3.9 in RHEL 8's RPM

RDBMS

- Fix #22926: `az mysql server create/update`: Update default value for mysql storage size

REST

- `az rest`: Support Unicode characters in JSON request body

Search

- `az search service create`: Add `--hosting-mode` argument to support S3HD SKU

Security

- `az security atm cosmosdb`: Add CLI support for ATP settings (Defender) on Cosmos DB

Service Connector

- `az webapp connection create`: Add `--private-endpoint` to support private endpoint connection
- `az spring connection create`: Remove client-type limitation

Service Fabric

- `az sf managed-cluster create`: Fix tag parsing for cluster command

SQL

- `az sql elastic-pool create`: Add support for HighAvailabilityReplica count for HS Elastic pools
- `az sql midb update`: Add update command

SQL VM

- `az sql vm update`: Add configuration options for SQL Assessment pre-requisites

Storage

- [BREAKING CHANGE] `az storage share close-handle`: Remove `--marker` which is not supported by sdk
- [BREAKING CHANGE] `az storage share snapshot`: Now only returns version, etag and last_modified info instead of all share properties
- `az storage account generate-sas`: Fix output sas random ordering for `srt` segment
- Fix #22563: `az storage blob upload`: Fix storage blob upload to a through pipe encode error
- Fix #20452: `az storage container policy create\update\list\show\delete`: Add new permissions, currently support `racwdxyltmei`

- Fix #22679: `az storage account file-service-properties update`: Fix `AttributeError: 'NoneType' object has no attribute 'smb'`
- Fix #22845: `az storage account genarate-sas`: Fix the flag `--auth-mode login` cause `AttributeError`

Synapse

- `az synapse sql pool create`: Add parameter `--collation`
- `az synapse link-connection`: New command group to support synapse link connections

May 24, 2022

Version 2.37.0

ACR

- Fix some `az acr manifest` commands do not correctly handle `-u/-p` credentials resulting in auth failure when not logged in to `az cli`
- Fix some `az acr` commands do not handle certain next-link tokens correctly resulting in exceptions when paging
- Fix some `az acr manifest` commands do not correctly parse some FQDNs resulting in exceptions

AKS

- [BREAKING CHANGE] `az openshift`: Remove the deprecated command group
- `az aks create`: Add new option `--node-resource-group` to specify the name of the resource group where user resources are stored
- `az aks get-credentials`: Raise exception when existing config file is invalid
- `az aks check-acr`: Add new option `--node-name` to specify the name of a specific node to perform acr pull test checks
- Fix #22032: `az aks nodepool add/update`: Fix autoscaler parameters for user node pools
- `az aks create/update`: Add Microsoft Defender security profile support
- GA Kubernetes version alias
- `az aks update`: Add support for updating kubelet identity with `--assign-kubelet-identity`

API Management

- Fix apim's `apply-network-updates` command

App Service

- Fix #18151: `az webapp config backup restore`: Fix the bug that 'WebAppsOperations' object has no attribute 'restore_slot'

ARM

- `az resourcemanagement private-link create`: Create Resource management private link
- `az resourcemanagement private-link delete`: Delete Resource management private link
- `az resourcemanagement private-link show`: Get Resource management private link
- `az resourcemanagement private-link list`: List Resource management private link
- `az private-link association create`: Create private link association
- `az private-link association delete`: Delete private link association
- `az private-link association show`: Get private link association
- `az private-link association list`: List private link association
- `az group delete`: Add new parameter `--force-deletion-types` to support force deletion
- `az bicep restore`: Add new command to restore external modules
- `az bicep build`: Add new parameter `--no-restore` to allow compilation without restoring external modules
- `az bicep decompile`: Add new parameter `--force` to allow overwriting existing Bicep files
- `az resource wait`: Fix `--created` keeps waiting even when `az resource show` returns "provisioningState": "Succeeded"

ARO

- `az aro create`: Add support for FIPS modules, host encryption, and disk encryption for master and worker nodes

Backup

- `az backup vault resource-guard-mapping`: Add support for updating, showing, and deleting ResourceGuardProxy
- Add multiple user authentication (MUA) support for critical operations: `az backup vault backup-properties set/az backup item set-policy/az backup policy set/az backup protection disable`
- Add `--tenant-id` parameter in critical commands: `az backup vault backup-properties set/az backup item set-policy/az backup policy set/az backup protection disable/az backup vault resource-guard-mapping` for cross-tenant scenario

Compute

- `az vm image list`: Add new server version aliases `Win2022AzureEditionCore` for offline list
- `az vm update`: Add additional license type SLES for `--license-type`
- `az vmss create`: Support enabling single placement group for Flexible VMSS
- `az disk create/update`: Add new parameter `--data-access-auth-mode` to support data access authentication mode
- `az sig show`: Add new parameter `--sharing-groups` to support query shared gallery group
- `az vm host group create`: Add new parameter `--ultra-ssd-enabled` to support Ultra SSD

Cosmos DB

- `az cosmosdb sql container update`: Fix bug to accept analyticalStorageTTL arg

Event Hubs

- `az eventhubs namespace schema-registry`: Add cmdlets for schema registry

Identity

- `az identity list-resources`: Add new command to support list the associated resources for identity

IoT

- `az iot dps policy` and `az iot dps linked-hub`: Fix DPS state updating

- `az iot central app private-link-resource list`: Add a new command to support listing private link resources
- `az iot central app private-endpoint-connection show`: Add a new command to support showing details of a private endpoint connection of the IoT Central app
- `az iot central app private-endpoint-connection approve`: Add a new command to support approving a private endpoint connection for the IoT Central app
- `az iot central app private-endpoint-connection reject`: Add a new command to support rejecting a private endpoint connection for the IoT Central app
- `az iot central app private-endpoint-connection delete`: Add a new command to support deleting a private endpoint connection for the IoT Central app

Key Vault

- Fix #22457: `az keyvault key decrypt/encrypt`: Fix returning bytes for `--output tsv`

Monitor

- [BREAKING CHANGE] `az monitor alert`: Deprecate whole command group, please use `monitor metrics alert`
- [BREAKING CHANGE] `az monitor autoscale-settings`: Deprecate whole command group, please use `az monitor autoscale`
- [BREAKING CHANGE] `az monitor activity-log list`: Deprecate parameter `--filters`.
- [BREAKING CHANGE] `az monitor activity-log list`: Deprecate parameter flag `--resource-provider`, please use `--namespace`

NetAppFiles

- `az netappfiles volumes export-policy add`: Fix `rule-index` validation and parameter made non required
- `az netappfiles ad add`: Add new optional parameter `site`
- `az netappfiles ad update`: Add new optional parameter `site`

Network

- `az network watcher connection monitor create`: Change for using user-provided workspace-ids even if output-type is missing
- `az network dns zone export`: Support traffic manager resources
- Private link add `Microsoft.Kusto/clusters` provider

- `az network lb create`: Add warnings for default SKU
- `az network lb address-pool`: Support connection draining on load balancer
- `az network application-gateway`: Add `settings`, `listener` and `routing-rule` command groups
- `az network application-gateway create`: Add parameter `--priority`
- `az network application-gateway probe`: Add parameter `--host-name-from-settings`
- [BREAKING CHANGE] `az network vnet peering`: Deprecate parameter flag `--remote-vnet-id`

Packaging

- Bump embedded Python to 3.10 for deb packages
- Use Mariner 2.0 GA image to build RPM

RDBMS

- `az mariadb server create/update`: Support `--minimal-tls-version`
- Change MySQL MemoryOptimized tier name to BusinessCritical

Reservations

- Update Reservation command with latest SDK

Role

- [BREAKING CHANGE] `az az/role`: Migrate the underlying API of `az ad` and `az role` from AD Graph API to Microsoft Graph API. For more details, see [Microsoft Graph migration](#)

Security

- `az security alerts-suppression-rule`: Add alerts suppression rules to security module

Service Bus

- `az servicebus queue update`: Fix message time to live
- `az servicebus queue`: Add `ReceiveDisabled` to `--status`

- `az servicebus namespace create/update`: Add `--disable-local-auth` to enable or disable SAS authentication
- `az servicebus namespace private-endpoint-connection/private-link-resource`: New command groups

Service Connector

- [BREAKING CHANGE] `az containerapp connection create`: Default `client_type` changed to `none`
- `az containerapp connection`: Add new command group to support container app connection
- `az containerapp connection create`: Add `--container` parameter in interactive mode
- `az spring connection`: Add support for `az sping-cloud` renaming Add new parameter key value pair to support password from KeyVault

Service Fabric

- `az sf cluster node-type add`: Fix the unexpected error that 'StorageAccountsOperations' object has no attribute 'create'

SQL

- Fix #22316: `az sql server ad-admin create`: Fix Display Name and Object ID to be required

SQL VM

- `az sql vm update`: Add configuration options for SQL Best Practices Assessment

Storage

- [BREAKING CHANGE] `az storage share show`: Remove `contentLength`, `hasImmutabilityPolicy` and `hasLegalHold` from the output result
- [BREAKING CHANGE] `az storage blob snapshot`: Now only returns version info instead of all blob properties
- Fix #21819: `az storage fs directory`: Add new command `generate-sas`
- `az storage account show-connection-string`: Append endpoints by default
- Fix #22236: `az storage entity insert`: Fix `--if-exists fail` not working

- `az storage copy`: Fix `--exclude-path` `TypeError`
- `az storage blob download`: Allow downloading to stdout for pipe support
- Fix #22209: `az storage entity insert`: Fix `Edm.Boolean` not working
- `az storage directory/file list`: Add `--exclude-extended-info` to exclude some properties info from response, default to `False`
- Fix #21781: `az storage blob upload/download`: Progress fix
- `az stroage entity query`: Fix UUID type is not JSON serializable
- `az storage blob delete-batch`: No longer exits after individual delete failure

April 26, 2022

Version 2.36.0

ACR

- `acr task run`: Add `--no-format` option
- `acr task logs`: Add `--no-format` option
- `acr taskrun logs`: Add `--no-format` option

AKS

- `az aks create`: Add `--nat-gateway-managed-outbound-ip-count` and `--nat-gateway-idle-timeout` to support nat gateway integration
- `az aks create`: Add `managedNATGateway` and `userAssignedNATGateway` to supported outbound type
- `az aks check-acr`: Bump canipull to 0.0.4-alpha to skip location check if cname returns only privatelink

AMS

- `az ams asset-track create`: Add command to create an asset track
- `az ams asset-track show`: Add command to show an asset track
- `az ams asset-track list`: Add command to list all tracks under an asset
- `az ams asset-track update`: Add command to update the parameters of a track
- `az ams asset-track update-data`: Add update-data command to refresh the server in case track file was updated
- `az ams asset-track delete`: Add command to delete track

- `az ams streaming-endpoint get-skus`: Add command to get skus under a streaming endpoint

App Config

- Fix feature flag import for missing description when using 'appconfig/kvset' profile

App Service

- `az staticwebapp create`: Allow creating Static Web Apps not connected to a github repo
- Fix #21943: `az webapp config backup create`: Fix AttributeError 'str' object has no attribute 'get'

Backup

- `az backup policy create/set`: Add support for creating/updating IaaSVM MBPD policy

Bot Service

- `az bot directline/email/facebook/kik/msteams/skype/slack/sms/telegram create`: Add `--location` argument as specified by user to channel creation for regionality/EUDB

CDN

- `az afd rule create`: Fix rule creation failure with action type RouteConfigurationOverride
- `az afd route create`: Fix route creation issue with disabled `--link-to-default-domain` option
- Fix #22066: `az cdn name-exists` missing type argument

Compute

- `az vm create`: Fix the bug of "NoneType object has no attribute lower" when creating Flex VMSS without `--vm-sku` parameter
- `az restore-point create`: Add a new parameter `--source-restore-point` to support cross region copy

- `az restore-point show`: Add a new parameter `--instance-view` to show the instance view of a restore point and replace the deprecated `--expand`
- `az restore-point collection show`: Add a new parameter `--restore-points` to show all contained restore points in the restore point collection and replace the deprecated `--expand`
- `az sig image-version create`: Add new parameter `--target-region-cvm-encryption` to support Confidential VM encrypting the OS disk
- `az vm/vmss create`: Install guest attestation extension and enable system managed identity by default when Trusted Launch configuration is met
- `az vm/vmss create`: Add new parameter `--disable-integrity-monitoring` to disable the default behavior (installing guest attestation extension and turning on MSI) when creating VM/VMSS compliant with Trusted Launch

IoT

- [BREAKING CHANGE] `iot dps access-policy`: Deprecate `access-policy` in favor of `policy`

Key Vault

- `az keyvault key`: GA SKR and keyvault key rotation
- Fix #20520: `az keyvault network-rule`: Support removing multiple IP

NetAppFiles

- `az netappfiles volume-group`: Add command group to manage volume group resources

Network

- Fix #21845: `az network routeserver create` required `--public-ip-address` argument
- Fix #21829: `az network traffic-manager endpoint update` required `--type` argument
- Private link add `Microsoft.Network/privateLinkServices` provider
- Fix #22085: `az network nsg rule create` has no attribute "is_default"

Packaging

- Release DEB package for Ubuntu 22.04 Jammy Jellyfish
- Release RPM package for RHEL 8, CentOS Stream 8
- Release RPM package for Mariner 1.0, 2.0 preview

RDBMS

- `az postgres server create`: Fix error message for invalid server names

Security

- Add `az security automation` CLI commands

Service Bus

- `az servicebus namespace create`: Add zone redundant parameter
- `az servicebus namespace authorization-rule keys renew`: Add `--key-value` parameter

Service Connector

- `az webapp connection`: Add command `create sql/webpubsub` to support more target resources

SQL

- `az sql mi create`, `az sql mi update`: Add `--service-principal-type` parameter to support Win Auth (Kerberos)

Storage

- Fix #21914: `az storage blob upload`: Make block size larger (100MB) for large files (>200GB)
- `az storage account/container/blob generate-sas`: Add `--encryption-scope`
- Fix #21920: `az storage copy` & `az storage remove`: Hide credentials in warning message
- Add `--blob-endpoint/--file-endpoint/--table-endpoint/--queue-endpoint` for data service commands to support customized service endpoint
- GA storage file datalake soft delete
- `az storage cors add`: Allow `PATCH` for `--methods`

- `az storage entity`: Support specifying `EdmType` for `--entity`
- Fix #21966: `az storage blob download-batch`: Fix failure when `--pattern` is blob name
- Fix #21414: `az storage blob sync`: Fix the flag `--delete-destination` default to `false`
- `az storage account blob-inventory-policy create`: Add missing fields, add `excludePrefix` in filter

April 05, 2022

Version 2.35.0

ACR

- [BREAKING CHANGE] `az acr create`: Reject request with a name using uppercase letters
- [BREAKING CHANGE] `az acr connected-registry create`: Reject request with a name using uppercase letters
- `az acr update`: Disable public network now displays a warning message
- Deprecate `az acr manifest metadata` command group
- `az acr manifest`: Add `show-metadata`, `list-metadata` and `update-metadata` commands

AKS

- `az aks create/update`: Add new parameters `--enable-gmsa`, `--gmsa-dns-server`, `--gmsa-root-domain-name` to support Windows gMSA v2
- `aks enable-addons`: Add new parameter `--enable-msi-auth-for-monitoring` to support enabling managed identity auth
- `az aks snapshot create`: Move to `az aks nodepool snapshot create`
- `az aks snapshot delete`: Move to `az aks nodepool snapshot delete`
- `az aks snapshot list`: Move to `az aks nodepool snapshot list`
- `az aks snapshot show`: Move to `az aks nodepool snapshot show`
- `az aks create`: Add `--pod-subnet-id` to support dynamically assigne pod ip
- `az aks nodepool add`: Add `--pod-subnet-id` to support dynamically assigne pod ip
- `az aks create`: Add `--kubelet-config` and `--linux-os-config` to support custom node configuration

- `az aks nodepool add`: Add `--kubelet-config` and `--linux-os-config` to support custom node configuration

AMS

- `az ams account identity assign`: Add ability to assign managed identity to media services account
- `az ams account identity remove`: Add ability to assign managed identity to media services account
- `az ams transform create`: Add new parameter `blur-type` for FaceDetector presets
- `az ams account encryption set`: Add new parameters `system-assigned` and `user-assigned` to allow users to set managed identities to their account encryption
- `az ams account storage set-authentication`: Add new parameters `system-assigned` and `user-assigned` to allow users to set managed identities for their storage account attached to Media Services

APIM

- `apim api schema create`: Add new command to support creating a schema for graphql API
- `apim api schema delete`: Add new command to support deleting the schema of an API
- `apim api schema list`: Add new command to support showing the list of schema's of an API
- `apim api schema show`: Add new command to support getting the schema of an API
- `apim api schema entity`: Add new command to support getting the schema entity tag
- Onboard to private endpoint for API Management

App Config

- `az appconfig kv export`: Stop throwing error if no key-values are exported to App Service
- `az appconfig create`: Add new options `retention-days` and `enable-purge-protection`
- `az appconfig list-deleted`: Add new command to list all deleted but not yet purged App Configurations

- `az appconfig show-deleted`: Add new command to show properties of a deleted but not yet purged App Configuration
- `az appconfig recover`: Add new command to recover a deleted but not yet purged App Configuration
- `az appconfig purge`: Add new command to purge a deleted store

App Service

- Fix #21439: `az webapp deploy`: Fix `--async` argument value in help message
- Fix #21574: `az webapp vnet-integration add`: Fix the AttributeError that 'NoneType' object has no attribute 'server_farm_id'
- `az staticwebapp create` : Change default output location and API location to `None`. Change default app location to "/". Remove unnecessary properties from output
- `az staticwebapp show` : Remove unnecessary properties from output
- `az staticwebapp list` : Remove unnecessary properties from output
- `az staticwebapp update` : Remove unnecessary properties from output
- `az webapp deployment slot create`: Allow overriding container settings
- Fix #21080: `az webapp up`: Fix object has no attribute 'response'
- Fix #19747: `az webapp up`: Fix TypeError: 'NoneType' object is not iterable
- `az webapp up`: Validate that ASE exists, is an ASE v3, and not an ILB ASE; Validate that preexisting plan is on the ASE; Default to I1V2 SKU if using an ASE
- Fix #20240: `az functionapp deployment source config-zip`: Fix the bug that the parameter `--slot` doesn't work
- Fix #12090: `az webapp create`: Allow plan in different resource group from web app
- `az staticwebapp identity assign`, `az staticwebapp hostname set`, `az staticwebapp create`: Fix #21186: Show detailed error message instead of "bad request"
- `az staticwebapp update`: Fix #21465: Allow specifying static web app resource group
- Fix #21728: `az webapp deployment github-actions add`: Allow passing in runtime with colon delimiter
- `az webapp config`: Fix for Web App Persistent Storage gets disabled after each deployment
- `az appservice ase create-inbound-services`: Add support for Azure private DNS zone creation in ASEv3

ARM

- Fix #20842: `az bicep`: Fix to use requests environment variables for CA bundle

- `az policy assignment create`: Support `--subscription` parameter

Backup

- List commands multi-page response bug fix
- `az backup restore restore-disks`: Add support for Original Location Restore and Alternate Location Restore
- `az backup policy create/set/list`: Add support for creating and selectively listing Enhanced policies
- `az backup protection enable-for-vm`: Add support for Trusted VM configure protection with Enhanced policies
- `az backup vault backup-properties`: Add new parameter `--hybrid-backup-security-features` to support setting the security features for hybrid backups

CDN

- Upgrade `azure-mgmt-cdn` to 12.0.0 for Azure Front Door Standard/Premium GA

Cognitive Services

- Upgrade to use API 2022-03-01
- Add new command `az cognitiveservices account list-models`

Compute

- [BREAKING CHANGE] `az vm/vmss create`: Remove the default value `contributor` of parameter `--role`
- `az vm host`: Add new command `restart` to support dedicated host reboot
- `az vm extension show`: Add new parameters `--instance-view` to support track the vm extension progress
- Change help info of `--enable-bursting` to flag it is for on-demand only
- Fix #20174: `az vm create`: Determine plan information when using image alias
- `az disk/snapshot/sig definition create/update`: Add new parameters `--architecture` to support ARM64
- `az vm disk attach`: Add new parameter `--disks` to support attaching multiple disks in one API call
- `az vm/vmss create`: Support creating VM/VMSS from community gallery image
- `az vm/vmss create`: Add community gallery legal agreement acceptance

- `az vm/vmss create`: Add the verification of whether `--os-type` is correct when creating VM from community gallery image or shared gallery image

Cosmos DB

- `az cosmosdb update`: Support updating key vault key uri
- `az managed-cassandra cluster update`: Allow `--external-seed-nodes`, `--external-gossip-certificate` and `--client-certificate` to take empty list
- `az managed-cassandra cluster`: Fix `--repair-enabled` as of type `three_state_flag`

Event Grid

- Fix #21521: System topic subscription update attribute error
- Support user identity and mixed mode

Event Hubs

- `az eventhub namespace update`: Fix disable eventhub capture and autoinflate

Key Vault

- Fix #18319 & #21555: `az keyvault list-deleted`: List all deleted resources if no specified resource type
- `az keyvault key create`: Support `--default-cvm-policy`
- Fix #21330: `az keyvault network-rule remove`: Fix ip address remove issue

NetAppFiles

- `az netappfiles snapshot restore-files`: New command to restore specified files from the specified snapshot to the active filesystem
- `az netappfiles volume create`: Add optional parameters `--enable-subvolumes`
- `az netappfiles volume delete`: Add optional parameter `--force-delete` or `--force`
- `az netappfiles volume update`: Add optional parameter `--unix-permissions`
- `az netappfiles subvolume`: New command group to manage subvolume resources
- `az netappfiles subvolume create`: New command to create subvolume
- `az netappfiles subvolume show`: New command to get specified subvolume
- `az netappfiles subvolume update`: New command to update specified subvolume

- `az netappfiles subvolume list`: New command to get all subvolume in a specified volume
- `az netappfiles subvolume delete`: New command to delete specified subvolume
- `az netappfiles subvolume metadata`: New command group to manage subvolume metadata resources
- `az netappfiles subvolume metadata show`: New command to get details about a specified subvolume
- `az netappfiles account ad add`: New optional parameters to support ldap search scope `--user-dn`, `--group-dn` and `--group-filter`
- `az netappfiles account ad update`: New optional parameters to support ldap search scope `--user-dn`, `--group-dn` and `--group-filter`

Network

- `az network nat gateway`: Validate attaching public IPs
- `az network lb`: Support inbound NAT rule port mapping query
- Fix #21716: `az network private-dns zone import`: Allow hyphenated SRV records
- `az network application-gateway waf-policy managed-rule exclusion rule-set`: Support pre-rule exclusion creation without exclusion

Packaging

- Use Red Hat Universal Base Image 8 to build `e18` RPM package
- Bump Python image to `3.10.3-alpine3.15`
- Bump MSI embedded Python to 3.10.3

RDBMS

- Fix operations.py file installing dependencies in CloudShell

Role

- [BREAKING CHANGE] `az ad sp create-for-rbac`: Stop defaulting `--scopes` to subscription
- [BREAKING CHANGE] `az ad sp create-for-rbac`: When creating a self-signed certificate in keyvault, `validity_months` is changed from `years * 12 + 1` to `years * 12`

Service Bus

- `az servicebus topic subscription rule create`: Add filter type parameter

Service Connector

- `az webapp/spring-cloud connection create/update`: Provide `--service-endpoint` parameter to support vnet scenario
- `az webapp/spring-cloud connection`: Add command `create redis/redis-enterprise` to support more target resources

SQL

- [BREAKING CHANGE] `az sql db tde list-activity`: Command no longer exists
- [BREAKING CHANGE] `az sql mi show/create/update/list`: Instead of `backupStorageRedundancy`, `currentBackupsStorageRedundancy` and `requestedBackupStorageRedundancy` properties are returned
- `az command sql db str-policy set`: Make `diffbackup_hours` parameter optional

Storage

- [BREAKING CHANGE] Fix #21494: `az storage blob upload/upload-batch`: Fix `--content-md5` for upload, ignore `--content-md5` for upload-batch
- [BREAKING CHANGE] `az storage table/entity`: `--timeout` is removed for all sub commands
- [BREAKING CHANGE] `az storage entity query/show`: `--accept` is removed
- `az storage table/entity`: Add `--auth-mode login` to support RBAC
- `az storage blob upload/upload-batch`: Make precondition work
- `az storage blob upload-batch`: No longer exits on the first failure
- Fix #21591: `az storage blob upload`: Fix storage blob upload not auto guessing file type
- Fix `az storage entity merge`: Stop automatically casting `DisplayVersion` to float
- `az storage blob download`: Support downloading managed disk with both SASUri and OAuth by specifying `--blob-url` with `--auth-mode login`
- Fix #21699: `az storage blob upload-batch`: Fix upload-batch result url truncation issue
- `az storage account\container\blob generate-sas`: Allow new permissions

Synapse

- `az synapse role assignment list`: Fix showing only 100 results
- `az synapse notebook import`: Fix `--folder-path` parameter problem

March 03, 2022

Version 2.34.1

App Service

- Hotfix: Fix #20489: `az webapp log tail`: Fix the AttributeError that 'NoneType' object has no attribute 'host_name_ssl_states'
- Hotfix: Fix #20747: `az webapp create-remote-connection`: Fix the EOFError that ran out of input
- Hotfix: Fix #20544: `az webapp config snapshot restore`: Fix the AttributeError that 'WebAppsOperations' object has no attribute 'restore_snapshot'
- Hotfix: Fix #20011: `az webapp config ssl bind`: Fix the AttributeError that 'str' object has no attribute 'value'
- Hotfix: Fix #19492: `az webapp config backup restore`: Fix the AttributeError that 'WebAppsOperations' object has no attribute 'restore'

Storage

- [BREAKING CHANGE] `az storage blob upload/upload-batch`: Fix `--overwrite` that it no longer overwrite by default

March 01, 2022

Version 2.34.0

ACR

- `az acr manifest`: Add new command group to support managing artifact manifests in Azure Container Registries
- Deprecate `az acr repository show-manifests` command and replace with `acr manifest metadata list` command

AKS

- `az aks nodepool update`: Add `--node-taints` to allow modify node taints
- `az aks get-credentials`: Add new parameter `--format` to support specifying the format of returned credential
- `az aks nodepool`: Allow specifying `--scale-down-mode` in nodepool create and update

APIM

- `az apim api import`: Update api-id description #18306
- Fix #21187: `az apim api create/update/import`: Fix header and query param names being swapped

App Config

- `az appconfig kv import`: Add new parameter `--strict` to support strict import

App Service

- [BREAKING CHANGE] `az webapp up`: Change supported runtimes
- [BREAKING CHANGE] `az webapp create`: Change supported runtimes
- [BREAKING CHANGE] `az webapp list-runtimes`: Add `--os / --os-type` argument, change runtimes, change default behavior to return both linux and windows stacks, and deprecate `--linux` argument
- [BREAKING CHANGE] `az functionapp create`: Take runtime names and versions from API instead of hardcoded list
- `az functionapp plan`: Update the max value of `--max-burst` to 100
- `az functionapp list-runtimes`: Add new command to show function app runtimes, versions, and compatible functions versions
- `az webapp create`: Provide support `--https-only` flag
- `az webapp deployment github-actions remove`: Fix the bug that path cannot start with a slash

ARM

- `az account management-group entities`: Add a new command group to support entities (Management Groups and Subscriptions) operations for the authenticated user

- `az account management-group hierarchy-settings`: Add a new command group to support operations on hierarchy settings defined at the management group level
- `az account management-group tenant-backfill`: Add a new command group to support backfilling subscriptions for the tenant
- `az account management-group subscription show`: Get the details of a given subscription under a given management group
- `az account management-group subscription show-sub-under-mg`: Show what subscription is under a given management group
- `az account management-group check-name-availability`: Check if a management group name is valid and available
- `az deployment`: Fix the bug of 'bytes object has no attribute get' for error handling in retry cases

Backup

- Add private endpoints support for Microsoft.RecoveryServices/vaults

Compute

- `az vm create`: Fix the issue that VMCustomization is not enabled
- `az vm disk attach`: Modify help description to guide how to use the `--ids` parameter correctly
- `az restore-point`: Add new command group to support managing restore point
- `az vmss create/update`: Add new parameters `--security-type`, `--enable-secure-boot` and `--enable-vtpm` to support Trusted Launch
- `az vmss create/update`: Add new parameters `--automatic-repairs-action` to support repair action
- `az vmss create/update`: Add new parameters `--v-cpus-available` and `--v-cpus-per-core` to support VMSize customization

Cosmos DB

- `az managed-cassandra cluster update`: Fix to allow `--external-seed-nodes` and `--external-gossip-certificates` to be updated by the user

Eventhub

- `az eventhubs namespace create`: Add `--user-assigned`, `--system-assigned`, `--encryption-config`

- `az eventhubs namespace identity`: Cmdlets for event hubs identity
- `az eventhubs namespace encryption`: Cmdlets for event hubs encryption
- `az servicebus namespace create`: Add `--user-assigned`, `--system-assigned`, `--encryption-config`
- `az servicebus namespace identity`: Cmdlets for event hubs identity
- `az servicebus namespace encryption`: Cmdlets for event hubs encryption

IoT

- `az iot hub create`: Add the `--enforce-data-residency` parameter to support creating resources with data residency enforced (and cross-region disaster recovery disabled)
- `az iot dps create`: Add the `--enforce-data-residency` parameter to support creating resources with data residency enforced (and cross-region disaster recovery disabled)

Key Vault

- Fix #21341: `az keyvault update`: Support updating tags
- `az keyvault key create/import/set-attributes`: Support `--immutable` to mark release policy immutable
- `az keyvault key import`: Support `--kty oct` to import AES key

Monitor

- `az monitor log-analytics workspace table`: Add new command `create`, `delete` and `search-job create` to support Microsoft/Custom log/Search Results table operations
- `az monitor log-analytics workspace update`: Add a new parameter `--data-collection-rule` to support update defaultDataCollectionRuleResourceId
- `az monitor log-analytics workspace table`: Add new command `restore create` and `migrate` to support Restored logs table/migrate operations

Network

- `az bastion ssh`: Provide support for Bastion SSH access on Darwin and Linux
- `az network private-endpoint`: Associate IP configurations and ASGs when creating PE

Packaging

- [BREAKING CHANGE] Drop Ubuntu 14.04 Trusty Tahr and Debian 8 Jessie support
- [BREAKING CHANGE] Drop Ubuntu 21.04 Hirsute Hippo support
- Add Ubuntu 21.10 Impish Indri support
- Bump embedded Python to 3.8 for deb packages

Profile

- [BREAKING CHANGE] `az account show`: Drop `--sdk-auth`

RDBMS

- Fix bug for private dns zone provisioning to vnet resource group in different subscription
- Enable `rdbms-connect` extension in Cloud Shell

Role

- Add warning to `role` and `ad` commands about Microsoft Graph migration

SQL

- `az sql server create/update`: Add federated client id support

Storage

- `az storage account create/update`: Support `--sam-account-name` and `--account-type`
- `az storage blob upload`: Add `--tier`, migrate to track2
- `az storage blob upload-batch`: Migrate to track2

February 14, 2022

Version 2.33.1

Compute

- Hotfix: Fix #21224: Fix the issue that VMCustomization is not enabled

Packaging

- [BREAKING CHANGE] Drop jmespath-terminal from docker image

February 01, 2022

Version 2.33.0

ACR

- `az acr connected-registry create`: Add `--notifications` to support adding patterns for generating notification events on connected registry artifacts
- `az acr connected-registry update`: Add `--add-notifications` and `--remove-notifications` to support adding or removing patterns for generating notification events on connected registry artifacts

AKS

- `az aks nodepool add/update/upgrade`: Add new parameter `--aks-custom-headers` to support custom headers
- `az aks create`: Add new parameter `--snapshot-id` to support creating a nodepool from snapshot when creating a cluster
- `az aks nodepool add/upgrade`: Add new parameter `--snapshot-id` to support creating a nodepool from snapshot
- `az aks snapshot create/delete/list/show`: Add new commands to support the management of snapshot related operations
- `az aks update/az aks nodepool update`: Allow empty string as label value

App Config

- [BREAKING CHANGE] Support app service slots

App Service

- `az webapp vnet-integration add`: Fix a bug that prevented adding a vnet in a different subscription from the webapp
- `az functionapp vnet-integration add`: Fix a bug that prevented adding a vnet in a different subscription from the functionapp
- `az webapp create`: Support joining a vnet in a different subscription

- `az functionapp create`: Support joining a vnet in a different subscription
- `az functionapp create`: Remove preview from PowerShell runtime for linux
- `az appservice plan update`: Add `--elastic-scale` and `--max-elastic-worker-count` parameters to support elastic scale
- `az webapp update`: Add `--minimum-elastic-instance-count` and `--prewarmed-instance-count` parameters to support setting instance count
- `az webapp up`: Add help text and debug text for configuration saving and loading
- `az webapp list-runtimes`: Support node 16-lts runtime for linux and windows

Batch

- `az batch create/activate`: Add clarify application package path help info for argument `--package-file`

Bot Service

- `az bot create`: Add location as specified by user to bot creation for regionality/EUDB

Compute

- `az image builder create`: Add new parameter `--proxy-vm-size` to support proxy VM size customization
- `az image builder create`: Add new parameter `--build-vm-identities` to support user assigned identities customization
- `az vmss update`: Add new parameter `--force-deletion` to support force delete VMSS
- `az vm/vmss create`: Add warning log and modify help to inform that the default value `Contributor` of `--role` will be removed
- `az disk-encryption-set create`: Make the parameter `--source-vault` un-required
- `az vm create/update`: Add new parameters `--v-cpus-available` and `--v-cpus-per-core` to support VMSize customization

Cosmos DB

- `az managed-cassandra cluster status`: Add table format support

Key Vault

- `az keyvault create`: Add default permissions on keyvault creation

Monitor

- `az monitor action-group`: Support event hub receiver

NetAppFiles

- `az netappfiles account ad add`: Add new optional parameter named encrypt-dc-connections
- `az netappfiles volume export-policy add`: Add missing optional parameters kerberos5_read_only, kerberos5_read_write, kerberos5i_read_only, kerberos5i_read_write, kerberos5_p_read_only, kerberos5_p_read_write, has_root_access, chown_mode
- `az netappfiles account ad update`: Add command

Network

- Add Microsoft.DataFactory/factories to supported Private Endpoints
- Add Microsoft.Databricks/workspaces to supported private endpoints
- `az network private-endpoint`: Add parameter and subgroup to support IP Configuration, ASG and NicName
- `az network traffic-manager endpoint create/update`: Add new arguments `--min-child-ipv4` and `--min-child-ipv6`.
- Add Microsoft.HybridCompute/privateLinkScopes to supported Private Endpoints

Packaging

- Update Dockerfile base image from Alpine 3.14 to 3.15

RDBMS

- `az postgres flexible-server create`: Change default postgres version

Redis

- `az redis create`: Add default value for identity and public network access as `None`

ServiceConnector

- Support new target resources: servicebus, eventhub, appconfig

Storage

- Stop supporting `--auth-mode login` for `az storage blob sync` and `az storage fs directory upload/download`

January 04, 2022

Version 2.32.0

AKS

- `az aks create`: Add new parameter `--enable-fips-image` to support enabling fips image
- `az aks nodepool add`: Add new parameter `--enable-fips-image` to support enabling fips image

App Service

- [BREAKING CHANGE] `az webapp up`: Remove support for the python|3.6 (linux and windows), ruby|2.5 (linux), and php|7.3 (windows) runtimes. Add support for the python|3.9 runtime (linux), php|8.0 (linux), and ruby|2.7 (linux)
- [BREAKING CHANGE] `az webapp create`: Remove support for the python|3.6 (linux and windows), ruby|2.5 (linux), and php|7.3 (windows) runtimes. Add support for the python|3.9 runtime (linux), php|8.0 (linux), and ruby|2.7 (linux)
- [BREAKING CHANGE] `az functionapp create`: Remove python 3.6 support
- Fix #19550: `az staticwebapp users update`: Allow updating static web app user roles again
- `az logicapp create`: Autogenerate a WS1 App Service Plan when no value for `--plan` or `--consumption-plan-location` is provided
- `az appservice plan create`: Allow creating App Service Plans for Logic Apps (SKUs WS1, WS2, and WS3)
- Fix #20757: `az webapp up`: Fix list index out of range when no `--plan` argument passed
- Fix #18652: `az webapp up`: Search for *.csproj in child directories
- `az webapp list-runtimes`: Remove support for the python|3.6 (linux and windows), ruby|2.5 (linux), and php|7.3 (windows) runtimes. Add support for the python|3.9 runtime (linux), php|8.0 (linux), and ruby|2.7 (linux)

Backup

- `az backup restore restore-azurewl`: Add client side validations
- `az backup container unregister`: Support MAB type for parameter `--backup-management-type`
- `az backup protectable-item list/show`: Add auto-protection policy and node-list field in the response for SQLInstance SQLAG
- `az backup protection auto-enable-for-azurewl/auto-disable-for-azurewl`: Add support for SQLAG

Compute

- `az vm/vmss create/update`: Expand validate license types for `--license-type` parameter
- `az sig image-definition list-shared`: Add new parameters `--marker` and `--show-next-marker` to support paging
- `az sig image-version list-shared`: Add new parameters `--marker` and `--show-next-marker` to support paging

IoT

- `az iot hub update`: Add error handling for file-upload parameters and fixes empty \$default storage endpoint errors
- `az iot central app create`: Add new parameter `--mi-system-assigned` to support creating an app with system-assigned managed identity
- `az iot central app identity show/assign/remove`: Add new commands to manage the system-assigned managed identity to an existing IoT Central app
- `az iot dps access-policy`: Be replaced with `az iot dps policy`
- `az iot dps linked-hub create`: Add convenience arguments for linking hubs

Network

- Fix #19482: Azure Bastion AAD fix for new CLI core changes
- `az network lb inbound-nat-pool create`: Add new parameter `--backend-pool-name`

Profile

- `az account show/set`: Add `-n`, `--name` argument

Redis

- `az redis identity`: Add support for assigning and modifying Identity

REST

- [BREAKING CHANGE] `az rest`: Remove `resourceGroup`, `x509ThumbprintHex` transforms

Role

- [BREAKING CHANGE] `az ad sp create-for-rbac`: Drop `name` property from the output. Use `appId` instead
- [BREAKING CHANGE] `az ad sp create-for-rbac`: No role assignment will be created by default

Storage

- `az storage copy`: Add positional argument `extra_options` to pass through options to `azcopy`

Synapse

- [BREAKING CHANGE] `az synapse managed private endpoints create`: Remove `--resource-id` and `--group-id`, use `--file` instead
- `az synapse sql pool create/restore`: Add parameters `--storage-type` to support specifying storage account type
- `az synapse kql-script`: New command group to support Kusto script

December 07, 2021

Version 2.31.0

AKS

- `az aks update`: Support edit nodepool label after creation
- `az aks nodepool update`: Support edit nodepool label after creation
- `az aks create`: Fix issue that `--attach-acr` parameter can't work

AMS

- Remove deprecated variable 'identifier_uri' from creating sp method
- Update api version for AMS and AVA private link registration

App Service

- `az functionapp create`: Add support for creating a webapp joined to a vnet
- `az webapp up`: Fix failure to detect dotnet 6.0 web apps
- `az appservice ase update`: Support for allowing new private endpoint connections on ASEv3
- `az appservice ase list-addresses`: Support ASEv3
- `az staticwebapp identity assign`: Assign managed service identity to the static web app
- `az staticwebapp identity remove`: Disable static web app's managed service identity
- `az staticwebapp identity show`: Display static web app's managed service identity
- Fix #17507: `az staticwebapp functions`: Add support for linking existing function app to static webapp (bring your own functions)
- `az staticwebapp create`: Update help text with guidance for repos in Github organizations
- `az functionapp deployment source config-zip`: Fix #12289: Allow build on zip deploy for windows function apps
- `az staticwebapp create`: Add better error message when attempting to create a static webapp that already exists
- `az appservice`: Fix AttributeError during user error handling
- `az appservice plan create`: Add `--zone-redundant` parameter to support enabling zone redundancy for high availability
- `az webapp ssh`: Add proxy support
- `az webapp create-remote-connection`: Add proxy support
- `az webapp log download/tail`: Add proxy support
- `az webapp create`: Fix container registry server url parsing for `--deployment-container-image-name/-i` argument
- `az functionapp deployment source config-zip`: Fix returning success when the deployment did not succeed
- `az staticwebapp appsettings set`: Make set functional
- `az staticwebapp appsettings`: Switch to the new SWA app settings SDK methods
- `az functionapp plan create`: Add `--zone-redundant` parameter to give the option to create a zone redundant app service plan

- Support managed identity in App Service container

ARM

- `az resource\group list`: Support querying data only by passing the tag name to `-tag` parameter
- `az account management-group`: Add new parameters `--no-register` to skip RP registration for `Microsoft.Management`
- `az deployment`: Prettify error output for ARM deployment
- `az bicep install`: Add a new parameter `--target-platform/-t` to specify the running platform of Bicep CLI
- `az bicep upgrade`: Add a new parameter `--target-platform/-t` to specify the running platform of Bicep CLI
- `az deployment sub/tenant/mg create`: Fix the `KeyError: 'resourceGroup'` in outputting results in table format when deploying non-resource group level resources
- `az policy assignment create` and `az policy assignment identity assign` support adding user assigned identity
- `az bicep install`: Work now behind a corporate proxy

Backup

- GA `az backup` and some bug fixes
- `az backup protectable-item list/show`: Fix AttributeError for `server_name`
- `az backup restore restore-disks`: Add support for Cross Zonal Restore

Cognitive Services

- `az cognitiveservices account deployment`: Add new commands `show`, `list`, `create`, `delete`
- `az cognitiveservices account commitment-plan`: Add new commands `show`, `list`, `create`, `delete`
- `az cognitiveservices commitment-tier`: Add new command `list`

Compute

- Fix #20182: `az snapshot create`: Fix auto-detection bug for `--copy-start`
- Fix #20133: `az vm create`: Fix `--data-disk-delete-option` not working when no `--attach-data-disks` are provided

- Fix boot diagnostics decoding
- `az vm create/update`: Add new parameter `--enable-hibernation` to support enabling hibernation capability
- `az vm/vmss run-command show`: Add new parameter `--instance-view` to support tracking the progress of RunCommand
- Update the help description for unmanaged disks
- `az disk create/update`: Add `--public-network-access` argument to control the policy for export on the disk
- `az disk create/update`: Add `--accelerated-network` argument to support the accelerated networking
- `az snapshot create/update`: Add `--public-network-access` argument to control the policy for export on the disk
- `az snapshot create/update`: Add `--accelerated-network` argument support the accelerated networking
- `az snapshot create`: Fix #20258: Fix creating a snapshot of a Uniform VMSS OS disk

EventGrid

- GA `az eventgrid system-topic`

Key Vault

- `az keyvault key encrypt/decrypt`: Support AES algorithm for MHSM
- `az keyvault key rotation-policy update`: Support both camel case and snake case json for `--value`

NetAppFiles

- `az netappfiles volume create`: Fix volume export policy

Network

- `az network express-route peering connection ipv6-config`: Add new commands `set`, `remove`
- `az network application-gateway waf-policy managed-rule exclusion`: Add new subgroup `rule-set` to support per rule exclusions
- `az network bastion create`: Fix invalid validator when `--scale-units` is None

- `az network vnet create`: Add `--enable-encryption` argument to support enable encryption on virtual network
- `az network vnet update`: Add `--enable-encryption` argument to support enable encryption on virtual network
- `az network vnet create`: Add `--encryption-enforcement-policy` argument to choose If Virtual Machine without encryption is allowed in encrypted Virtual Network.
- `az network vnet update`: Add `--encryption-enforcement-policy` argument to choose If Virtual Machine without encryption is allowed in encrypted Virtual Network.

Packaging

- Support Python 3.10
- Add Dockerfile.mariner to support Mariner build

Profile

- `az logout`, `az account clear`: Remove ADAL token cache file `accessTokens.json`

RDBMS

- Fix private DNS zone suffix bug
- Fix #20124: `az mysql/postgres flexible-server db create`: Make resource group and server name required
- `az postgres flexible-server`: Remove preview tag

Storage

- `az storage share list-handle/close-handle`: New commands for share handle
- GA account level and blob version level immutable storage

Synapse

- [BREAKING CHANGE] `az synapse sql/pool audit-policy`: Remove `--blob-auditing-policy-name`
- `az synapse notebook/spark-job-definition`: Add `--folder-path` argument
- `az synapse spark pool create/update`: Add `--spark-config-file-path`
- `az synapse spark job submit`: Fix for `--main-class-name`

- `az synapse sql-script`: New command group to support sql script management

November 02, 2021

Version 2.30.0

Core

- [BREAKING CHANGE] Migrate from ADAL to MSAL. For more details, see [MSAL-based Azure CLI](#)

ACR

- [BREAKING CHANGE] `az connected-registry`: `--repository` flag short version `-t` is being removed.
- [BREAKING CHANGE] `az connected-registry install renew credentials`: Now it requires the user to confirm password generation.
- `az connected-registry install`: Deprecate and redirect to `az acr connected-registry get-settings`.
- `az connected-registry repo`: Deprecate and redirect to `az acr connected-registry permissions update`.
- `az connected-registry permissions show`: A new command that allows the user to see the sync scope map information.
- `az connected-registry get-settings`: A new command that retrieves the necessary information to install a connected registry and allows the generation of a new sync token password.
- `az connected-registry create`: No longer adds a postfix to the sync token and scope map name.

AKS

- `az aks create/update`: Add new parameter `--aks-custom-headers` to support for custom headers
- `az aks create`: Support setting `--private-dns-zone` to none for private cluster creation
- `az aks create/update`: Add new parameter `--enable-secret-rotation` and `--rotation-poll-interval` to support secret rotation
- `az aks enable-addons`: Add new parameter `--enable-secret-rotation` and `--rotation-poll-interval` to support secret rotation

App Config

- `az appconfig kv import/export`: Add new parameter `--profile` to support using `appconfig/kvset` profile

App Service

- Fix #19617: `az webapp ssh`: Open Web SSH on the specified instance
- `az staticwebapp hostname`: Support adding static webapp hostname via TXT validation
- Enable support for PowerShell on Linux function apps with V4

ARM

- `az bicep publish`: Add new command to publish bicep modules

ARO

- `az aro create`: Remove Identifier URIs

Compute

- `az disk update`: Fix the problem that updating network access policy to `AllowPrivate` failed
- `az vm update`: Add `--host` argument and `--host-group` argument to support assign an existing VM to a specific ADH
- Fix #19599: `az vm create`: Fix the issue that `--nic-delete-option` not working when no `--nics` is provided.
- `az snapshot create`: Support copyStart as createOption
- `az vmss create/update`: Support in-guest patching for VMSS
- `az vm application set/list`: Add new commands to support VM application
- `az vmss application set/list`: Add new commands to support VMSS application
- `az vm create`: Add `--ephemeral-os-disk-placement` argument to support choosing the Ephemeral OS disk provisioning location
- `az vmss create`: Add `--ephemeral-os-disk-placement` argument to support choosing the Ephemeral OS disk provisioning location
- `az vm update`: Add `--size` argument to support the resize
- `az vmss update`: Add `--vm-sku` argument to support the resize

- `az vm run-command`: Add new commands to support managing the running commands in VM
- `az vm update`: Add `--ephemeral-os-disk-placement` argument to support choose the Ephemeral OS disk provisioning location
- `az vmss update`: Add `--ephemeral-os-disk-placement` argument to support choose the Ephemeral OS disk provisioning location
- `az sig gallery-application`: Add new commands to support managing gallery application
- `az sig gallery-application version`: Add new commands to support managing gallery application version
- GA the features related to Flex VMSS

Container

- `az container create`: Add parameter `--zone` to support Availability Zone selection
- `az container create`: Fix the issue that `--subnet` or `--vnet` cannot be used with IP address type `Public` to allow `Private`
- `az container create`: Add Support for `--registry-login-server` to work with `--acr-identity`

Cosmos DB

- `az cosmosdb mongodb retrieve-latest-backup-time`: Add new command for fetching latest restorable timestamp for Mongo Account.
- `az cosmosdb locations`: Add new commands for listing account locations and their properties.
- `az managed-cassandra cluster/data-center`: GA support for managed cassandra cluster and data center

DMS

- `az dms project create/az dms project task create` : Add MySQL projects/tasks for offline migrations.

FunctionApp

- [BREAKING CHANGE] `az functionapp devops-pipeline`: Remove commands and move them to `functionapp` extension

HDInsight

- `az hdinsight create`: Add two parameters `--zones` and `--private-link-configurations` to support creating cluster with availability zones feature and creating private link enabled cluster with private link configurations feature.

Key Vault

- Support Keyvault SKR
- `az keyvault key random`: Request some random bytes from managedHSM
- `az keyvault rotation-policy/key rotate`: Support rotate key and manage key rotation policy
- `az keyvault create/update`: Add `--public-network-access` parameter

Monitor

- `az monitor metrics alert condition` : Add support for 'skip metric validation'

NetAppFiles

- [BREAKING CHANGE] `az netappfiles account backup-policy create/update`: Remove optional parameter `--yearly-backups`.
- `az netappfiles account list`: Add option to skip `--resource-group` parameter and fetch accounts for subscription.
- `az netappfiles pool create`: Add optional parameter named `--encryption-type`
- `az netappfiles volume create`: Add optional parameters: `--network-features`, `--avs-data-store`, `--default-group-quota`, `--default-user-quota`, `--is-def-quota-enabled`
- `az netappfiles volume update`: Add optional parameters: `--default-group-quota`, `--default-user-quota`, `--is-def-quota-enabled`

Network

- `az network bastion create`: Add new parameter `--scale-units` and `--sku` to support setting scale unit
- `az network vnet`: Add parameter `--bgp-community`
- `az network private-endpoint-connection`: Support "Microsoft.Cache/Redis"
- `az network private-endpoint-connection`: Support "Microsoft.SignalRService/WebPubSub"

RDBMS

- Introduce MySQL georestore command and update validators
- GA `az mysql flexible-server`

Service Bus

- Fix MU capacity to include 16 when updating namespace

ServiceConnector

- `az webapp/spring-cloud connection`: New command group to support service to service connection

SQL

- `az sql server ad-admin`: Fix breaking change made to update and delete

Synapse

- `az synapse kusto`: Add Kusto pool(mgmt) support

October 29, 2021

Version 2.29.2

ARO

- Hotfix: `az aro create`: Remove Identifier URIs

October 21, 2021

Version 2.29.1

Compute

- Hotfix: Fix static webapp commands that are broken due to the upgrade of `azurerm` `mgmt-web` to 4.0.0

October 12, 2021

Version 2.29.0

AKS

- `az aks check-acr`: Bump canipull to 0.0.3 alpha to support sovereign cloud
- `az aks create/update`: Add new parameter `--disable-local-accounts` to support disable local accounts
- `az aks enable-addons`: Support open-service-mesh addon
- `az aks create/update`: Add support for updating tags

App Config

- Fix dependencies for multiple installations of `jsondiff` and `javaproPERTIES`

App Service

- `az webapp create/up`: Correct the typo of wrong java version in help
- `az logicapp create/delete/show/list`: Add new commands to support logicapp related operations
- `az staticwebapp environment delete`: Add command to support deleting static app environment
- `az functionapp show`: Add kind validation for show operation
- `az webapp config backup list`: Fix issue that returned backup configuration instead of backup list
- `az logicapp start/restart/stop`: Add new commands for logicapp
- `az webapp config storage-account`: Update parameter descriptions

ARM

- `az deployment`: Remove the log of printing request body from custom policy
- `az deployment group create`: Fix incorrect scope in the example of creating deployment from template-spec
- `az ts create`: Simplify overwrite confirmation message

Backup

- `az backup container register`: Fix refresh container bug

- `az backup`: Add CRR functionality for Azure Workload
- `az backup`: Add support for MAB backup management type in some sub commands

Compute

- `az sig create/update`: Add new parameter `--soft-delete` to support soft delete
- `az sig image-version`: Add new parameter `--replication-mode` to support setting replication mode
- `az vm/vmss update`: Fix disassociation VM/VMSS from capacity reservation
- `az vm/vmss create`: Hide alias `--data-delete-option` in help
- `az vmss create`: Support quick creation for flexible VMSS

Container

- [BREAKING CHANGE] `az container create`: Remove `--network-profile` parameter, property no longer supported
- `az container logs`: Fix the attribute error introduced by Track 2 migration
- `az container create`: Add parameter `--acr-identity` for support of MSI authenticated ACR image pull

Cosmos DB

- `az cosmosdb identity assign/remove`: Add support for user identity

Eventhub

- `az eventhubs namespace update`: Add `--infra-encryption` for encryption (enable-require-infrastructure-encryption).
- `az eventhubs namespace create/update`: Add `--disable-local-auth` to enable or disable SAS authentication.
- `az eventhubs namespace`: Add `private-endpoint-connection` and `private-link-resource` command groups

Key Vault

- [BREAKING CHANGE] Fix #18479: `az keyvault network-rule add`: Fix the bug which allows duplicate `--ip-address` with the ones already in the network-rule

- Fix #10254: `az keyvault network-rule add`: Add capability to accept multiple ip-addresses as a list in the form of `--ip-address ip1 [ip2] [ip3]...`
- `az keyvault delete`: Add warning when deleting managed HSM

Network

- Add `az network custom-ip-prefix wait`
- Add `az network vnet-gateway packet-capture wait`
- Add `az network vnet-gateway vpn-client ipsec-policy wait`
- Add `az network vnet-gateway nat-rule wait`
- Add `az network vpn-connection packet-capture wait`
- Private link and endpoint support for provider `Microsoft.BotService/botServices` to supported private endpoints operations
- `az network application-gateway client-cert`: Add commands `update` and `show`
- `az network application-gateway ssl-profile`: Add commands `update` and `show`
- `az network application-gateway http-listener create`: Add parameter `--ssl-profile`
- `az network application-gateway http-listener update`: Add parameter `--ssl-profile`
- Onboard hdinsight private link2 network cmdlets
- `az network bastion create`: Add `--tags` argument
- Private link and endpoint support for provider `Microsoft.Authorization/resourceManagementPrivateLinks`
- Private link and endpoint support for provider `Microsoft.MachineLearningServices/workspaces`

Profile

- `az account show`: Deprecate `--sdk-auth`

RDBMS

- [BREAKING CHANGE] `az postgres flexible-server migration`: Change `--properties @{filepath}` to `--properties {filepath}`
- `az postgres flexible-server migration create`: User can pass in filename with double quotes or no quotes and same for absolute paths.
- `az postgres flexible-server migration check-name-availability`: Add a command to check if a migration name is available.

- `az postgres flexible-server migration update`: Add `--start-data-migration` to reschedule the migration to start right now.
- Update list-skus, create command location setting and replica command

Role

- `az ad sp create-for-rbac`: Deprecate `--sdk-auth`

Security

- Add command `az security setting update`

Storage

- Fix #19279: Add clarification for file system name to also mean container name.
- Fix #19059: Fix doc link to point to public doc website
- `az storage account hns-migration start/stop`: Support migrate a storage account to enable hierarchical namespace
- `az storage container-rm create/update`: Add `--root-squash` to support enable nfsv3 root squash or all squash
- Fix #17858: `az storage blob upload`: make `--name` optional
- `az storage account create/update`: Add `--public-network-access` parameter
- `az storage container immutability-policy create`: Add `--allow-protected-append-writes-all/-w-all` parameter
- `az storage container legal-hold set`: Add `--allow-protected-append-writes-all/-w-all` parameter
- `az storage account create/update`: Enable account level immutability

Synapse

- [BREAKING CHANGE] `az synapse sql/pool audit-policy update`: Add parameter `blob-storage-target-state`, `log-analytics-target-state`, `event-hub-target-state` (at least choose one of these 3 paras)
- `az synapse integration-runtime`: Support start/stop integration-runtime
- `az synapse trigger`: Add az synapse trigger wait
- `az synapse trigger-run`: Add az synapse trigger-run cancel
- `az synapse integration-runtime`: Deprecate `create` command and will redirect to `managed create` or `self-hosted create` command

- `az synapse dataset/pipeline/linked-service/trigger`: Deprecate `set` command and will redirect to `update` command
- `az synapse workspace-package`: Support workspace package CRUD
- `az synapse spark pool update`: Support add or remove specific packages
- `az synapse workspace create/update`: Add arguments for supporting synapse workspace repository configuration
- `az synapse spark-job-definition`: Support spark job definition CRUD

September 09, 2021

Version 2.28.1

ARM

Hotfix: Fix #19468: pip installs azure-cli 2.0.73 because of the dependency on deprecated package `jsmin`

September 07, 2021

Version 2.28.0

ACR

- `az acr create/update`: Add support for disabling export through `--allow-exports`
- `az acr`: Bump core api-version to `2021-06-01-preview` from `2020-11-01-preview`. agent_pool, tasks and runs operations unchanged from `2019-06-01-preview`
- `az acr task credential`: Fix the issue where task credentials were not used
- `az acr task logs`: Fix the AttributeError when querying the task logs

AKS

- [BREAKING CHANGE] `az aks nodepool update`: Change rejecting the ability to use max-surge with node-image-only
- `az aks install-cli`: Add support for kubelogin darwin/arm64 releases
- Fix incorrectly passed parameter for option `--assign-kubelet-identity` in aks create sub-command
- Upgrade api-version to `2021-07-01` for ACS module
- `az aks create/update`: Add support for private cluster public fqdn feature

- Revert PR #18825: `az aks create/update`: Add parameter `--auto-upgrade-channel` to support auto upgrade (with fix)
- `aks create/aks nodepool add`: Add parameter `--os-sku` to support choosing the underlying container host OS

App Config

- `appconfig kv import/export`: Add endpoint validation during import and export

App Service

- `az webapp config storage-account list/add/update/delete`: Remove preview flag
- Fix #18497: `functionapp identity show`: Fix the crashes when the functionapp name does not reference an existing functionapp
- `az webapp config set`: Add additional help examples for powershell users
- Fix #17818: `az functionapp update`: Add instance validation for updating functionapp
- `az webapp config hostname add`: Fix the issue caused by AttributeError
- `az webapp config hostname add`: Fix the issue caused by AttributeError
- Fix #16470: `az staticwebapp secrets`: Add commands to manage deployment secrets
- `az webapp deployment source config-local-git`: Fix the issue caused by AttributeError when slot option is specified
- `az webapp deleted restore`: Fix the issue that 'WebAppsOperations' object has no attribute 'restore_from_deleted_app'
- `az webapp up`: Add ability to deploy Linux and Windows webapps to the same resource group
- `az webapp up`: Add support for deploying to an App Service Environment
- Fix #19098: `az webapp deployment slot auto-swap` : Fix the AttributeError error for parameters `--slot --disable`

ARM

- `az feature registration`: Add az feature registration apis
- `az tag create`: Add the note for handling existing tag in help
- `az ts create`: Fix issue where creating a template spec with inner deployments that reference a common template fails

CDN

- `az cdn endpoint create`: Fix endpoint creation failure with `--content-types-to-compress`

Compute

- `az ssh vm`: Raise error for managed identity and Cloud Shell
- Upgrade api-version for VM and VMSS from `2021-03-01` to `2021-04-01`
- `az vmss create/update`: Support spot restore policy to VM scale sets
- Add new examples for creating disk from share image gallery
- `az vm image list/list-offers/list-skus/list-publishers/show`: Add new parameter `--edge-zone` to support querying the image under edge zone
- Fix the issue caused by the lack of `os_type` when creating VM from shared gallery id
- Update shared image gallery doc
- `az capacity reservation`: Add new commands to manage capacity reservation
- `az capacity reservation group`: Add new commands to manage capacity reservation group
- `az vm create/update`: Add new parameter `--capacity-reservation-group` to support association to capacity reservation
- `az vmss create/update`: Add new parameter `--capacity-reservation-group` to support association to capacity reservation
- `az vmss create`: Support creating VMSS from shared gallery image

IoT

- `az iot hub/dps certificate update/create`: Add `--verified` argument to mark certificates as verified without proof-of-possession flow
- `az iot hub create/update`: Add `--disable-local-auth`, `--disable-device-sas`, and `--disable-module-sas` arguments to configure accepted SAS key authentication methods.

Key Vault

- `az keyvault private-endpoint-connection list`: Support list mhsm's private endpoint connections
- `az keyvault set-policy`: `--key-permissions` add new option `release`

Network

- Fix NSG rule creation example mistake
- Add a new command group `az network custom-ip prefix`.
- `az network public-ip`: Add parameter `--ip-address`.
- `az network public-ip prefix create`: Add parameter `--custom-ip-prefix-name`.
- `az network dns record-set {record-type} add-record`: Support idempotent
- PrivateLink supports `Microsoft.Purview/accounts` 2021-07-01
- `az network bastion ssh`: connect to a Virtual machine through ssh using Bastion Tunneling.
- `az network bastion rdp`: connect to a Virtual machine through native RDP using Bastion Tunneling.
- `az network bastion tunnel`: connect to a Virtual machine using Bastion Tunneling.

Packaging

- Use Python 3.9 in Homebrew formula
- When installed with RPM, run `python3.6` if available
- Add Ubuntu 21.04 Hirsute Hippo support
- Add Debian 11 Bullseye support
- Drop Ubuntu 20.10 Groovy Gorilla support

PowerBI

- Add private link provider `Microsoft.PowerBI/privateLinkServicesForPowerBI`

RDBMS

- [BREAKING CHANGE] `az postgres flexible-server migration`: Rename `--migration-id` to `--migration-name`
- [BREAKING CHANGE] `az mysql flexible-server create/update`: `--high-availability` available parameter is changed from 'Enabled' to 'ZoneRedundant' and 'SameZone' .
- Fix maintenance window update issue with MySQL and Change restart parameter to be case insensitive
- `az mysql flexible-server restore` enables network option change from private network to public network and vice versa.
- `az mysql flexible-server replica create`: Add `zone` parameter.

Role

- `az role assignment create`: Support `ForeignGroup` for `--assignee-principal-type`
- `az role assignment create`: Do not invoke Graph API if `--assignee-principal-type` is provided

SQL

- `az sql mi update`: Add `--subnet` and `--vnet-name` parameters to support the cross subnet update SLO
- Fix the enum name change in track2 Python SDK

Storage

- Fix #10765: Refine error message when account key is incorrect padding

Synapse

- [BREAKING CHANGE] Rename `az synapse workspace key update` to `az synapse workspace key activate` and remove `--is-active`
- Optimize submit spark job arguments
- `az synapse`: Add managed private endpoints feature.
- Spark pool remove library requirement

August 23, 2021

Version 2.27.2

Cosmos DB

- Hotfix: `az cosmosdb restore`: Fix the restore command for deleted accounts

August 17, 2021

Version 2.27.1

ARM

- Hotfix: Fix #19124: `az deployment what-if`: Handle unsupported and no effect change types

Batch

Upgrade batch data-plane to [azure-batch 11.0.0](#) ↗ Upgrade batch management-plane to [azure-batch-mgmt 16.0.0](#) ↗ `az batch location`: Add `list-skus` command to list SKUs available in a location `az batch account`: Add `outbound-endpoints` command to list outbound network dependencies

August 03, 2021

Version 2.27.0

ACR

- [BREAKING CHANGE] `az acr connected-registry install info`: Add a new required parameter `--parent-protocol`.
- [BREAKING CHANGE] `az acr connected-registry install renew-credentials`: Add a new required parameter `--parent-protocol`.
- `az acr import`: Support new parameter `--no-wait`
- Fix the Python SDK compatibility issue when migrating Track 2
- `az acr build`: Make file `.dockerignore` include directories with !

AKS

- `az aks check-acr`: Fix issues parsing certain client minor versions

AppConfig

- [BREAKING CHANGE] `appconfig feature set`: Set the value of parameter `--description` to empty string if it is not specified
- [BREAKING CHANGE] `az appconfig feature`: Support namespacing for feature flags and change output fields
- `az appconfig create`: Add tags support when creating resource

App Service

- `az webapp config set`: Add support for VNet Route All property.

- `az webapp vnet-integration add`: Default to VNet Route All. Allow cross subscription integration.
- `az appservice ase create`: Support for ASEv3 External and Zone redundancy
- `az webapp hybrid-connection add`: Improve help/error message and unblock Linux
- `az webapp config access-restriction remove`: Fix #18947 issue removing service endpoint rules
- : Fix #17424: `az appservice plan show`: Provide correct exit status

ARM

- `az what-if`: Fix output formatting
- `az bicep uninstall`: Add new command to uninstall bicep
- `az bicep build`: Fix an issue where running with --stdout doesn't print any output
- `az provider register`: Add deprecate info for `--accept-term`
- `az lock create/delete`: Add examples for operating different levels of locks
- `az deployment group/sub/mg/tenant create`: Add --what-if parameter for invoking What-If with the deployment create commands.
- `az deployment group/sub/mg/tenant create`: Add --proceed-if-no-change parameter to skip confirmation when --confirm-with-what-if is set and there's no changes in What-If results.
- Bump api-version from 2020-10-01 to 2021-04-01
- `az ts create`: Make parameter `--template-file` support bicep file
- `az resource create`: Add example for creating site extension to web app
- `az ts export`: Fix the issue that export template specs with no linked templates failed

Backup

- `az backup vault`: Add support for Customer Managed Keys(CMK)
- `az backup restore restore-disks`: Add MSI usage in IaaS VM Restore

CDN

- `az cdn endpoint rule`: Add OriginGroupOverride action support

Compute

- `az sig image-version create`: Support mixing disks, snapshots, and vhd
- `az vmss update`: Upgrade package version to fix securityProfile issue

- `az vm boot-diagnostics get-boot-log`: Fix crash when getting boot diagnostics log
- `az vm list-skus`: Fix the issue that it can't query the SKU which with partially zones available
- `az vm auto-shutdown`: Fix the issue that `--webhook` is required when `--email` is passed in
- `az vm create`: Support creating VM from shared gallery image
- `az vm secret add`: Add note to use Azure Key Vault VM extension instead in help

Container

- `az container exec`: Fix and improve terminal experience

DataBoxEdge

- Migrate databoxedge to track2 SDK

DMS

- `az dms project create/az dms project task create`: Remove MySQL projects/tasks for online migrations since they are no longer supported.

IoT

- `az iot hub create/update`: Add checks to prevent bad file-upload identity parameters when hub doesn't have identity
- `az iot hub create/update`: Add `--fileupload-notification-lock-duration` parameter
- `az iot hub create/update`: Deprecate `fileupload-storage-container-uri` parameter
- `az iot dps/hub certificate create`: Certificates will now always be uploaded in base64 encoding.

Key Vault

- [BREAKING CHANGE] Fix #13752: `az keyvault create` not idempotent. Creating existing keyvault will fail.
- Fix #6372: table output for secrets isn't correct

Maps

- `az maps creator create`: Support maps creator create managed
- `az maps creator update`: Support maps creator update managed
- `az maps creator list`: Support maps creator list managed
- `az maps creator show`: Support maps creator show managed
- `az maps creator delete`: Support maps creator delete managed

NetAppFiles

- `az netappfiles volume pool-change`: Update help description for pool-change

Network

- `az network application-gateway create`: Add `--ssl-certificate-name` argument
- Private link add Microsoft.ServiceBus/namespaces provider
- `az network application-gateway waf-policy custom-rule match-condition add`: Add examples
- `az network express-route port link update`: Add `--macsec-sci-state` argument.
- Private link add Microsoft.Web/hostingEnvironments provider
- `az network lb frontend-ip update`: Support cross tenant for argument `--gateway-lb`.
- `az network nic ip-config update`: Support cross tenant for argument `--gateway-lb`.
- Private link add Microsoft.StorageSync/storageSyncServices provider
- Private link add Microsoft.Media/mediaservices provider
- Private link add Microsoft.Batch/batchAccounts provider

Packaging

- Add licenses to all Python packages
- Add SOCKS Proxy Support

PolicyInsights

- Migrate to track 2 SDK

RDBMS

- PostgreSQL, MySQL migration to GA API

Redis

- `az redis create\update`: Add new parameter `--redis-version`

SQL

- Update Microsoft.Sql to track2 SDK
- `az sql server outbound-firewall-rule create`: Azure CLI Commands for Outbound Firewall Rules

Storage

- Fix #18352: `az storage fs file list --exclude-dir` breaks with `--show-next-marker`
- `az storage fs generate-sas`: Support generate sas token for file system in ADLS Gen2 account
- `az storage account blob-service-properties`: Support last access tracking policy
- `storage container-rm migrate-vlw`: Support Version level Worm (VLW)
- `az storage copy` add new option `--cap-mbps`

Synapse

- `synapse workspace key update`: Fix the issue that updating a workspace key failure due to parameter `--is-active-cmk` lost
- Reimport notebook failure

July 14, 2021

Version 2.26.1

ACR

- Hotfix: `az acr build\connected-registry\pack\run\scope-map`: Fix the compatibility bug caused by SDK upgrade

AKS

- Hotfix: `az aks create`: Fix the issue that `assign-kubelet-identity` option can't work

Storage

- Hotfix: Fix issue caused by jwt upgrade.
- Hotfix: `az storage fs directory download`: Fix the issue with `--sas-token` to generate valid sas url
- Hotfix: `az storage blob copy start`: Fix the issue in copy from different account

July 06, 2021

Version 2.26.0

AKS

- Migrate ACS module to track 2 SDK
- Upgrade api-version to 2021-05-01 for ACS module
- Add UltraSSD support
- Support use custom kubelet identity
- `az aks get-credentials`: Add a check for KUBECONFIG environmental variable

APIM

- Add version parameter for apim api import
- Fix apim upgrade bug when specifying protocols
- `az apim create`: Fix `--enable-managed-identity` true failure

App Config

- Stop overwriting KeyVault reference content type during import

App Service

- [BREAKING CHANGE] `az functionapp create`: Remove support for EOL Node 8 and 10
- [BREAKING CHANGE] `az webapp deployment source config`: Remove vsts-cd-manager
- [BREAKING CHANGE] `az functionapp deployment source config`: Remove vsts-cd-manager
- `az webapp/functionapp config access-restriction add`: Prevent duplicate rules using service endpoints.

- `az webapp/functionapp config access-restriction remove`: Remove service endpoints are case-insensitive
- `az webapp config access-restrictions add`: Skip validation if user does not have access to get service tag list.
- Add support for Linux Consumption and improve how content share name is generated.
- : Fix an issue where adding VNET integration & Hybrid connections on a slot is not working
- `az appservice domain create`: Fix get correct domain agreements
- `az webapp deployment github-actions add/remove`: new commands

AppConfiguration

- Add support for `disable_local_auth`

ARM

- `az provider register`: Make parameter `--accept-term` become not required

ARO

- `az aro create`: Add cidr values for pod/service
- Fail if resource doesn't exist on delete

Azurestack

- Azure Stack Hub Support for AKS and ACR has been added in 2020-09-01-hybrid profile

Backup

- `az backup container`: Fix container registration Workload container registration fix, SDK upgraded to 0.12.0, Fixed and Re-ran tests
- Add Archive Support for Azure CLI

Billing

- Migrate billing to track2 SDK

Cognitive Services

- `az cognitiveservices account`: Add list-deleted, show-deleted, recover, purge commands

Compute

- `az sig create/update`: Add --permissions to specify the permission of sharing gallery.
- `az sig share`: Manage gallery sharing profile.
- `az sig list-shared`: List shared galleries by subscription id or tenant id.
- `az sig show-shared`: Get a shared gallery.
- `az sig image-definition list-shared`: List shared galleries by subscription id or tenant id.
- `az sig image-definition show-shared`: Get a shared gallery image.
- `az sig image-version list-shared`: List shared galleries by subscription id or tenant id.
- `az sig image-version show-shared`: Get a shared gallery image version.
- `az vmss create`: Support NetworkApiVersion for Vmss with OrchestraionMode == Flexible
- Make dependent resources of VM/VMSS support edge zone
- Update from CoreOS to Flatcar
- Add the hint to suggest users use the standard public IP when creating VM

Container Registry

- Migrate to track2 SDK

Cosmos DB

- Add point-in-time restore commands to the stable branch.
- Add support for selecting Cosmos DB analytical storage schema type

HDInsight

- `az hdinsight create`: Remove the incoming breaking change notice for the parameter `--workernode-size` and `--headnode-size`.
- Add three new cmdlets to support new azure monitor feature:

NetAppFiles

- `az netappfiles account ad add`: Optional parameter added named --administrators
- `az netappfiles pool create`: Optional parameter added --cool-access
- `az netappfiles volume create`: Optional parameters added named --chown-mode, --cool-access, --coolness-period, --coolness-period
- `az netappfiles volume backup restore-status`: Command added to see backup restore status

Network

- `az network routeserver create`: Add --public-ip-address argument.

RDBMS

- Add autogrow parameter for MySQL and add database name to output json when created

Resource

- Third-party S2S Consent/Permission Enumeration

Security

- Remove preview from security module

SQL

- Bump sdk version
- Fix for server create in SQL 0.28
- `az sql db ledger-digest-uploads`: Support SQL Ledger
- Fix for IdentityType for UMI
- `az sql db str-policy set/show`: Add Set and Show ShortTermRetentionPolicy

Storage

- GA support secured SMB
- `az storage account create`: Support --enable-nfs-v3 to set NFS 3.0 protocol

- Support container soft delete

June 15, 2021

Version 2.25.0

ACR

- `az acr connected-registry`: Minor bug fixes

App Service

- `az webapp deployment source config-local-git`: Fix to set SiteConfig

ARM

- `az resource tag`: Fix the problem of tagging resources with resource type `Microsoft.Network/publicIPAddresses`
- `az policy assignment non-compliance-message`: New command group for policy assignment non-compliance messages
- `az policy assignment update`: New command for partially updating existing policy assignments

Backup

- Migrate backup to track2 SDK

Compute

- Upgrade api-version for VM and VMSS from '2020-12-01' to '2021-03-01'
- `az vm create`: Support delete option for NICs and Disks for VMs in Azure CLI
- Support user_data for VM and VM Scale Sets

Container

- `az container exec`: Decode received bytes as utf-8 string

EventGrid

- Migrate track2 SDK

HDInsight

- Migrate to track2 Python SDK 7.0.0

IoT Hub

- Fix for user-assigned identity ARM issue on remove

Key Vault

- Fix #11871: AKV10032: Invalid issuer error for operations in nondefault tenant/subscription
- `az keyvault set-policy/delete-policy`: Support `--application-id`
- `az keyvault recover`: Support MHSM
- `az keyvault private-link-resource list`: Support MHSM
- `az keyvault private-endpoint-connection`: Support MHSM

NetAppFiles

- `az netappfiles volume backup status`: Command added to get the status of the backup for a volume.
- `az netappfiles volume update`: Optional parameter added named `--snapshot-policy-id` to assign a snapshot policy to the volume.
- `az netappfiles volume backup create`: Optional parameter added named `--use-existing-snapshot` to manually backup an already existing snapshot.
- `az netappfiles volume backup update`: Optional parameters added named `--use-existing-snapshot` to manually backup an already existing snapshot. Optional parameter `label` also added to add a label to backup.

Network

- Support `Microsoft.Sql/servers` provider in Private link
- `az network private-link-resource list`: Support `--type microsoft.keyvault/managedHSMs`
- `az network private-endpoint-connection`: Support `--type microsoft.keyvault/managedHSMs`

RDBMS

- Add commands for Github actions
- `az postgres flexible-server migration`: Add customer facing feature to migrate postgres db servers from Sterling to Meru platform
- Private DNS zone parameter added for restore command, high availability validator
- Change server default location (issue reported)

Role

- [BREAKING CHANGE] `az ad sp create-for-rbac`: `--name` is now only used as the `displayName` of the app. It is not used to generate `identifierUris` anymore. `name` in the output is now the same as `appID` (`servicePrincipalNames`) and deprecated.

SignalR

- `az signalr identity`: Add managed identity related command
- `az signalr cors update`: Add update command for cors

Storage

- `az storage blob copy start`: Support `--tier` and `--rehydrate-priority`
- GA release storage file share NFS and SMB multichannel
- [BREAKING CHANGE] `az storage account create`: Remove `StorageFileDataSmbShareOwner` option for `--default-share-permission`
- `az storage blob list`: `--delimiter` parameter value will now be honored

Synapse

- Update to AZ Synapse mgmt 2.0.0
- Spark configuration conversion, which cause the failure

Webapp

- Add to `az webapp deploy` param help text

June 02, 2021

Version 2.24.2

Container

- Hotfix: Fix #18276: `az container create` fails with `AttributeError: 'ResourcesOperations' object has no attribute 'create_or_update'`

June 01, 2021

Version 2.24.1

App Service

- Hotfix: Fix #18266 - webapp config appsettings set command causing all values to default to "false"

ARM

- Hotfix: Fix deserialization issue in the What-If formatter of ARM template

Compute

- Hotfix: Fix the bad request issue when creating VMSS in Azure Stack

IoT

- Hotfix: Fix issue for removing last user-assigned identity from IoT Hub

May 25, 2021

Version 2.24.0

AKS

- `az aks check-acr`: Add the nodeselector linux to avoid the "canipull" pod to be scheduled on the windows node
- Sdk update
- `az aks create` and `update` azure-rbac
- Add run-command cli

App Config

- Allow importing key-values with unicode characters from file

App Service

- [BREAKING CHANGE] `az webapp list-runtimes`: Add Dotnet6 support and update runtimes
- `webapp log tail`: Fix #17987: logging.warning call with invalid 'end' argument
- Fix #16838- az cli update app setting command always making slotsetting to true
- `az appservice`: Add function to retrieve users github personal access token
- `az staticwebapp appsettings set` issue #17792
- Fix #18033: az staticwebapp appsettings set of missing positional param `app_settings`
- Fix issues with APIs signature that changed with Track2 update
- Fix get resource management client properly
- Add interactive way to get token for staticwebapp
- Fix an issue where assign and remove identities would fail with a call to NoneType

ARM

- Migrate resource to track2 SDK
- `az ts`: Add UiFormDefinition file support to TemplateSpecs for GA (05/04)

ARO

- Add cluster credential rotation

Compute

- `az sshkey create`: Save private key to local file system

Cosmos DB

- Create and manage Role Definitions and Role Assignments for enforcing data plane RBAC on Cosmos DB SQL accounts

DevTestLabs

- `az labs create environment`: Fix error creating an environment from an ARM template

HDInsight

- [BREAKING CHANGE] `az hdinsight create`: Use getting default sku api to set workernode and headnode size if customer does not provide.

IoT

- `az iot hub create`: Support assigning identities and assigning roles to system-managed identity.
- `az iot hub update`: New parameter `--file-upload-storage-identity` to allow for managed-identity authenticated file upload.
- `az iot hub identity assign`: New command to assign user/system-assigned managed identities to an IoT Hub.
- `az iot hub identity show`: New command to show identity property of an IoT Hub.
- `az iot hub identity show`: New command to update identity type of an IoT Hub.
- `az iot hub identity remove`: New command to remove user/system-assigned managed identities from an IoT Hub.
- `az iot hub routing-endpoint create`: New `--identity` parameter allows choosing a user/system-assigned identity for routing endpoints.
- `az iot hub route create`: New routing source-type `DeviceConnectionStateEvents`

Kusto

- Update command group long summary

Network

- Bump api version from '2020-11-01' to '2021-02-01'
- New command group `az network lb address-pool tunnel-interface`
- `az network lb frontend-ip update`: New parameter `--gateway-lb`
- `az network nic ip-config update`: New parameter `--gateway-lb`
- `az network rule create/update`: New parameter `--backend-pools-name`
- `az network vnet-gateway create`: Add new paramter `--nat-rule`
- Add new cmd group `az network vnet-gateway nat-rule`
- `az network vpn-connction create`: Add new paramter `--ingress-nat-rule` and `--egress-nat-rule`
- `az network vnet create`: Add new parameter `--flowtimeout`

Packaging

- Support Python 3.9

RDBMS

- Change IOPS logic for MySQL
- Prevent private DNS zone track2 migration breaking rdbms module

Service Fabric

- [BREAKING CHANGE] `az sf cluster certificate`: Remove all commands under this group. Please follow the instructions in [Add a secondary certificate using Azure Resource Manager](#) to add/remove cluster certificates.
- [BREAKING CHANGE] `az sf managed-service update`: Remove deprecated parameter `--drop-source-replica-on-move`.
- [BREAKING CHANGE] `az sf managed-service create`: Remove deprecated parameters `--service-dns-name`, `--drop-source-replica-on-move` and `-instance-close-delay-duration`.
- [BREAKING CHANGE] `az sf cluster`: Rename parameter `--vault-resource-group` to `--vault-rg`.
- `az sf managed-cluster` and `sf managed-node-type`: Set groups as not preview
- Update `azure-mgmt-servicefabricmanagedclusters` package to the latest version 1.0.0 that uses 2021-05-01 GA api version.
- `az sf managed-cluster create`: Add parameters `--upgrade-mode`, `--upgrade-cadence` and `--code-version`.
- `az sf managed-node-type`: Add parameters `--data-disk-type`, `--is-stateless` and `--multiple-placement-groups`.

SQL

- `az sql server create`: Add a space to split the concatenated words in the help message of the argument `--assign-identity`.
- `az sql server update`: Add a space to split the concatenated words in the help message of the argument `--assign_identity`.

Storage

- [BREAKING CHANGE] `az storage share-rm delete`: Raise error when there are snapshots for target file share and add `--include` to specify deleting target file share and its snapshots
- `az storage blob generate-sas`: Add spaces to split the concatenated words in the help message of the arguments `--cache-control`, `--content-disposition`, `--content-encoding`, `--content-language` and `--content-type`.
- `az storage blob url`: Add a space to split the concatenated words in the help message of the argument `--snapshot`.
- `az storage container generate-sas`: Add spaces to split the concatenated words in the help message of the arguments `--cache-control`, `--content-disposition`, `--content-encoding`, `--content-language` and `--content-type`.
- Upgrade storage API version to 2021-04-01
- Support default share permission
- Support cross tenant object replication
- GA blob inventory
- `az storage share-rm list`: Support list with snapshots.

May 06, 2021

Version 2.23.0

ACR

- `az acr check-health`: Add support to verify dns routings to private endpoints
- Fix #17618: Update credential add/update handling for tasks created using `--auth-mode`

AKS

- `az aks update`: Add `--windows-admin-password` to support updating Windows password
- `az aks update`: Support updating from SPN cluster to MSI cluster.
- `az aks create`: Add `--enable-encryption-at-host` parameter

App Service

- [BREAKING CHANGE] Update websites SDK to the latest version (`azure-mgmt-web==2.0.0`) & Adopt track2 SDK
- [BREAKING CHANGE] Rename `az staticwebapp browse` to `az staticwebapp show`

- Add option of sku for `az staticwebapp create --sku`
- Add command `az staticwebapp update`
- `az webapp/functionapp config access-restriction add/remove`: Support for Service Tag, Http headers and multi-source rules.

ARM

- `az bicep`: Replace datetime APIs that are not available in Python 3.6
- `az deployment group create`: Fix the compatibility issue of api-version for parameter `--template-specs`

Backup

- `az backup vault create`: Add tags as an optional argument
- Make AFS configure backup flow idempotent

CDN

- `az cdn endpoint rule add`: Fix delivery rule creation for non-Microsoft SKU

Compute

- Extended location for Compute RP
- `az sig image-version create`: Support creating from a VHD
- `az vm create --count`: Support vnet and subnet configuration
- `az vmss extension upgrade`: Fix a bug
- Add error message for `vm identity assign`
- Zone-redundant storage (ZRS) managed disks
- `az disk create`: Trusted launch
- `az disk create`: Hibernation
- Fix a compatibility issue of old API version
- `az sig image version create`: Support data disk VHDs

Feedback reference

- Do not minify feedback issue body

FunctionApp

- Fix issue with zip deploy where local time was provided but UTC was expected
- Update stacks api json to add PowerShell on Linux in Functions

HDInsight

- Add Incoming BREAKING CHANGE for removing default value of `--workernode-size` and `--headnode-size`

Key Vault

- [BREAKING CHANGE] Support soft-delete feature for managed-HSM. `keyvault delete --hsm-name` will perform soft delete on a MHSM.

Marketplace Ordering

- New command group `az term` to accept/show terms

Misc.

- Define theme for Cloud Shell

Monitor

- New command `az monitor metrics list-namespaces`

Network

- [BREAKING CHANGE] az network dns record-set a show: Property `arecords` in output will be changed to `aRecords`.
- New command `az network express-route list-route-tables-summary`.
- New command `az network express-route peering get-stats`.
- New command `az network express-route peering connection list`.
- `az network lb create`: Add new parameter `--edge-zone`
- `az network nic create`: Add new parameter `--edge-zone`
- `az network private-endpoint create`: Add new parameter `--edge-zone`
- `az network private-link-service create`: Add new parameter `--edge-zone`
- `az network public-ip create`: Add new parameter `--edge-zone`
- `az network public-ip prefix create`: Add new parameter `--edge-zone`
- `az network vnet create`: Add new parameter `--edge-zone`

- New Command `az network lb list-nic`
- `az network application-gateway show-backend-health`: support probe operation arguments.
- `az network vpn-connection list`: support parameter `--vnet-gateway`.
- New command `az network vnet-gateway disconnect-vpn-connections`.
- New command `az network vnet-gateway vpn-client show-health`.
- New command `az network vnet-gateway vpn-client ipsec-policy show`.
- New command `az network vnet-gateway vpn-client ipsec-policy set`.
- New command `az network vnet-gateway packet-capture start`.
- New command `az network vnet-gateway packet-capture stop`.
- New command `az network vnet-gateway show-supported-devices`.
- New command `az network vpn-connection list-ike-sas`.
- New command `az network vpn-connection packet-capture start`.
- New command `az network vpn-connection packet-capture stop`.
- New command `az network vpn-connection show-device-config-script`.
- `az network private-link-resource list`: support more providers for `--type`

Packaging

- Bump python to `3.8.9` in docker image
- Bump bundled python to `3.8.9` in MSI.

RDBMS

- [BREAKING CHANGE] `az mysql flexible-server create`: `--storage-size` default value is changed from 10 to 32.
- `az postgres flexible-server create`: Add `--private-dns-zone` parameter for creating server with private access.

Role

- `az role assignment create/update`: Auto complete `assignee_principal_type`

SQL

- `az sql db create`: Add `--ha-replicas` argument
- `az sql db replica create`: Add `--ha-replicas` argument
- Allow short mw policy names for mi

SQL VM

- Make SqlServerLicenseType as optional

Storage

- Fix #16272 & #16853: Refine error message
- `az storage account create`: Add edge zone support
- Support user assigned identity for storage account
- `az storage account create/update`: Support sas&key policy

Synapse

- `az synapse notebook create`: Create a notebook

April 19, 2021

Version 2.22.1

ARM

- Hotfix: Fix the issue that bicep build broken in Python 3.6

Key Vault

- Hotfix: GA for managed-HSM ralated commands and parameters

April 13, 2021

Version 2.22.0

ACR

- [BREAKING CHANGE] `az acr connected-registry install info`: Replace keys ACR_REGISTRY_NAME, ACR_SYNC_TOKEN_NAME, ACR_SYNC_TOKEN_PASSWORD, ACR_PARENT_GATEWAY_ENDPOINT, and ACR_PARENT_PROTOCOL with a new connected string key, ACR_REGISTRY_CONNECTION_STRING.
- [BREAKING CHANGE] `az acr connected-registry install renew-credentials`: Replace keys ACR_REGISTRY_NAME, ACR_SYNC_TOKEN_NAME,

ACR_SYNC_TOKEN_PASSWORD, ACR_PARENT_GATEWAY_ENDPOINT, and ACR_PARENT_PROTOCOL with a new connected string key, ACR_REGISTRY_CONNECTION_STRING.

- `az acr connected-registry create`: Verify before the creation of the token and sync scope map that all ancestors are active.
- `az acr connected-registry create`: Add the repository and gateway permissions required for creation to all the ancestors of the new connected registry if needed prior to the connected registry creation.
- `az acr connected-registry delete`: Remove the gateway permissions of the deleted resources from all its ancestors' sync scope maps.
- `az acr connected-registry repo`: New command to add repository permissions to a connected registry and all its ancestors' sync scope maps, and remove repository permissions from the connected registry and all its descendants' sync scope maps

AKS

- `az aks create`: Add support for `--private-dns-zone` and `--fqdn-subdomain` feature

App Config

- Configure max line width for YAML parser to stop wrapping output
- Fix bug in print preview of restore command

App Service

- Fix #17219: Fix ssl bind bug
- Remove preview flag for Python 3.9 in create function app command
- Bugfix: Handle if only single publish profile is returned
- Fix #16203: az webapp log tail supports webapps running on Linux.

ARM

- [BREAKING CHANGE] `az bicep build`: Change the parameter `--files` to `--file`
- [BREAKING CHANGE] `az bicep decompile`: Change the parameter `--files` to `--file`
- Fix #17379: bicep auto install results in invalid json output from deployment
- `az bicep build`: Add a parameter `--outdir` for specifying the output directory
- `az bicep build`: Add a parameter `--outfile` for specifying the output file path

- Fix an issue where checking version upgrade for Bicep CLI throws exception if GitHub API rate limit is hit
- `az policy exemption`: Add new commands to support policy exemption

Backup

- Fix #14776: Fix `--force` parameter functionality for `az backup vault delete` command
- Fix on demand backup
- `az backup protectable-item list`: Add optional parameter `--backup-management-type`
- Fix policy create with rgNamePrefix and rgNameSuffix
- `az backup protectable-item list`: Add `--server-name` as an optional argument

Compute

- `az ssh vm`: Support VM SSH with Service Principal
- Add VMSS Rolling Upgrade opt
- New command: `vm install-patches`
- Disk encryption set: Add `--enable-auto-key-rotation`

Container

- Fix #16499: `az container create`: Fix handling of return value from `network_profiles.create_or_update`

Cosmos DB

- Support for managed service identity & default identity

EventGrid

- `az eventgrid system-topic create/update`: Add MSI Support
- `az eventgrid [partner topic | system-topic] event-subscription`: Add support for StorageQueueMessageTTL, AdvancedFilters, EnableAdvancedFilteringOnArrays
- `az eventgrid [partner topic | system-topic] event-subscription`: Add support for delivery attribute
- `az eventgrid topic create`: Add support for creating topic for azure or azurearc

Interactive

- Fix #16931: Fix `KeyError` in `az interactive --update`

NetAppFiles

- `az netappfiles account ad add`: Optional parameter added named `allow-local-ldap-users`
- `az netappfiles volume create`: Optional parameter added named `ldap-enabled`
- `az netappfiles volume backup status show`: Operation added
- Update backup tests

Network

- `az network vnet-gateway`: `--vpn-auth-type` allow multi value

Packaging

- [BREAKING CHANGE] RPM installed az now uses `python3` instead of hard-coded `/usr/bin/python3`.

RDBMS

- Allow DB server private access from different subscription
- Modify server create with private network, fix restore time bug

Search

- `az search service create`: Add `async (--no-wait)` options.
- `az search service update`: Add `async (--no-wait)` options.
- `az search shared-private-link-resource create`: Add `async (--no-wait)` options.
- `az search shared-private-link-resource update`: Add `async (--no-wait)` options.

Service Fabric

- Add managed application cli commands

Storage

- `az storage fs directory upload/download`: Support adls gen2 file system directory upload&download
- `az storage fs file list`: Support --show-next-marker
- `az storage share-rm`: Support create/show/delete snapshots

Synapse

- [BREAKING CHANGE] `az synapse role assignment create`: Role names at old version are not allowed, Sql Admin, Apache Spark Admin, Workspace Admin
- [BREAKING CHANGE] `az synapse role assignment create`: When --assignee argument can't uniquely determine the principal object, the command will raise error instead of adding a role assignment for the uncertain principal object.
- `az synapse role scope list`: List all scopes synapse supports.
- `az synapse role assignment create/list/delete`: Add --scope/--item-type/--item arguments to support manage role assignments based on scope.
- `az synapse role assignment create/list/delete`: Add --assignee-object-id argument, it will bypass Graph API and uniquely determine principal object instead of deducing principal object using --assignee argument.

March 23, 2021

Version 2.21.0

ACR

- Output a trace in `az acr login` for self-diagnosing potential docker command latency
- Fix #17172: When run check-health behind corporate proxy
- `acr update`: Support anonymous pull
- Fix #16700: Use "exists" api to check storage blob existence

AKS

- `aks update`: Add `--no-uptime-sla`
- Fix cross-sub assigning identity error and attach acr error
- Add support for node public IP prefix ID

APIM

- [BREAKING CHANGE] `apim backup`: `--storage-account-container` not support multi-value.
- [BREAKING CHANGE] `apim restore`: `--storage-account-container` not support multi-value.

App Service

- [BREAKING CHANGE] Fix #16087: `az webapp config ssl create`: set `--name` parameter as required.
- Fix #17053: `az webapp show` return null values for SiteConfig properties
- Fix #17207: `az webapp log config`: 'level' always defaults to verbose

ARM

- `az bicep build`: fix an issue where build warnings are not shown

Backup

- Add `id_part` for sub-resource names to fix `--ids`
- Fix #17094: Created separate test suite for CRR tests
- `az backup protection check-vm`: Add `--vm` and `--resource-group` as optional params

Cache

- GA `az cache`

CDN

- `az afd rule create`: Fix `--help` message

Compute

- Fix a Windows vm user update bug
- Fix #16585: `az vmss deallocate`: `--instance-ids` failed
- `az vm create`: New parameter `--platform-fault-domain` in FLEX VMSS mode
- `az vm create`: `--patch-mode` for Linux VM
- `az ssh vm`: Automatically launch browser when getting certificate fails
- `az vm create`: New parameter `--count`

- `az vm create`: Trusted Launch
- Fix #16037: `az vm open-port` accepts list of ports

Extension

- Add actionable message when an extension is not compatible with the CLI core

Key Vault

- `az keyvault role definition list`: Support `--custom-role-only` to list only custom role definitions
- Support keyvault custom role definition
- Add `--no-wait` for command `az keyvault security-domain download` and `--target-operation` for command `az keyvault security-domain wait`

NetAppFiles

- `az netappfiles account backup show`: Operation added.
- `az netappfiles account backup delete`: Operation added.
- `az netappfiles account ad add`: Parameter `--ldap-over-tls` added.
- `az netappfiles account create`: Parameter `--encryption` added.
- `az netappfiles account update`: Parameter `--encryption` added.
- `az netappfiles volume create`: Parameter `--encryption-key-source` added.
- `az netappfiles volume create`: Default export policy removed for nfsv4.1 and optional parameters added for setting up an export policy for nfsv4.1: `rule_index`, `unix_read_only`, `unix_read_write`, `cifs`, `allowed_clients`

Network

- `az network public-ip prefix create`: Support `--zone 1 2 3`
- `az network lb frontend-ip create`: Support `--zone 1 2 3`
- Bump version from '2020-08-01' to '2020-11-01'
- `az network lb address-pool`: Support subnet when creating or updating an IP-based backend pool of a load balancer.

RDBMS

- Added tests for flexible server team pipeline
- Python SDK migration

- Added PostgreSQL database create, show, and delete feature
- Updating Python SDK to 8.1.0b2

Role

- `az ad app permission list/grant`: Refine error message when no associated Service Principal exists for the App

Search

- `az search`: GA

Service Fabric

- `az sf certificate`: deprecate cluster cert commands.

SQL

- Add Server Trust Group commands

Storage

- Fix #16917: `az storage account generate-sas` fails if a connection string is provided
- Fix #16979: `az storage container create` fails when providing storage container metadata

Upgrade

- Fix #16952: Fix ImportError after upgrade

Misc.

- Allow configuring theme

March 02, 2021

Version 2.20.0

AKS

- Add support for SGX addon 'confcom'

AMS

- Update module to use 2020 Azure Media Services api.
- `az ams account encryption`: New subgroup to show or set the encryption for the media service account
- `az ams account storage set-authentication`: New command to set the authentication for the storage account associated with the media service account
- `az ams account create (mi-system-assigned)`: New --mi-system-assigned parameter for account create to set the managed identity of the media account
- `az ams account mru set`: This command will no longer work for Media Services accounts that are created with the 2020-05-01 version of the API or later.
- `az ams live-event create (stretch-mode, key-frame-interval, transcrip-lang, use-static-hostname, custom hostname)`: Add new parameter options to live-event create command
- `az ams live-event standby`: New command to put the live event in standby mode
- `az ams transform create (videoanalysismode, audioanalysis mode)`: New parameter options for transform create

App Service

- `az webapp config ssl bind`: handle if webapp and appservice plan in different rg.
Also reference text updates
- Fix #8743: `az webapp deploy`
- Bugfix: Add generateRandomAppName.json to setup
- `az functionapp create`: Add preview support for creating dotnet-isolated apps.
- Fix #12150: Support for subnet ID in vnet-integration add
- `az functionapp create`: Remove preview flag from Node.js 14.

ARM

- `az deployment group/sub/mg/tenant validate/create/what-if`: Add support for Bicep files
- `az bicep install`: New command for installing Bicep CLI
- `az bicep upgrade`: New command for upgrading Bicep CLI
- `az bicep build`: New command for building Bicep files

- `az bicep version`: New command for showing the current installed version of Bicep CLI
- `az bicep list-versions`: New command for showing the available Bicep CLI versions
- `az managedapp definition update`: Add new command for updating managedapp definition

Backup

- `az backup recoverypoint show-log-chain`: Add start/end time in show-log-chain table output
- BugFix: Enable Alternate Location Restore for SQL/SAPHANA protected items

CDN

- Add cli support for AFD SKU

Compute

- `az vm (extension) image list`: Make it more robust
- `az vmss create`: Fix a license type issue
- Upgrade API version to 2020-12-01
- `az vm create`: add `--enable-hotpatching`

Cosmos DB

- Upgrade to version 3.0.0 and add support for NetworkAclBypass + Update Mongo ServerVersion + backup policy

Extension

- Support config of extension index url

IoT Central

- `az iot central app`: Address several S360 fixes
- `az iot central app update`: Remove the need of checking etag when updating the existing iotc app.
- Change the resourceType (iotApps) to be in camel case.

Key Vault

- [BREAKING CHANGE] `az keyvault role assignment/definition list`:
`roleDefinitionName` should be `roleName` in command output
- [BREAKING CHANGE] `id` changes to be `jobId`, `azureStorageBlobContainerUri` changes to be `folderUrl` in command output of `az keyvault backup/restore`, `az keyvault key restore`

Network

- Bump version from '2020-07-01' to '2020-08-01'
- `az network public-ip create`: Support '--zone 1 2 3' after '2020-08-01'
- `az network routeserver peering`: Rename `--vrouter-name` by `--routeserver`
- `az network express-route peering create`: Support ipv6 address
- `az network public-ip create`: Expose a new argument `--tier`

OpenShift

- Update of `az openshift` deprecation warning

Search

- `az search`: Fix the `--identity-type` helper's guide.

SQL

- Update `az sql mi` examples
- `az sql db/elastic-pool create/update`: Add `maintenance-configuration` argument
- `az sql db replica create`: Add `--secondary-type` argument

Storage

- [BREAKING CHANGE] `az storage account file-service-properties`: Default to enable delete retention policy with retention days 7 in server side
- Fix #16872: `az storage blob` now (2.19) requires login even if `connection-string` is provided
- Fix #16959: `az storage copy` crashes: `ValidationError: local variable 'service' referenced before assignment`
- Fix #14054: 'NoneType' object has no attribute 'name'

- Fix #16679: `az storage blob download` fails with "Permission denied" if the destination file is a directory
- Upgrade storage api version to 2021-01-01
- Support version in Lifecycle management policy
- Support storage account shared key access management
- `az storage account network-rule`: GA resource access rules
- Support double encryption for encryption scope
- `az storage account blob-service-properties update`: Support --change-feed-retention-days
- Support rewrite existing blob

February 10, 2021

Version 2.19.1

Key Vault

- Hotfix: Dependency package `azure-keyvault-administration` is pinned to 4.0.0b1

February 09, 2021

Version 2.19.0

ACR

- `az acr connected-registry install info`: Add new key `ACR_SYNC_TOKEN_NAME` with the same value as `ACR_SYNC_TOKEN_USERNAME`. A warning that the latter will be deprecated is displayed.
- `az acr connected-registry install renew-credentials`: Add new key `ACR_SYNC_TOKEN_NAME` with the same value as `ACR_SYNC_TOKEN_USERNAME`. A warning that the latter will be deprecated is displayed.

AKS

- Add managed cluster stop/start bindings
- `az aks check-acr`: Fix Kubernetes version check

APIM

- GA the command group

App Config

- [BREAKING CHANGE] `az appconfig feature filter add`: Support adding JSON objects as feature filter parameter values

App Service

- `az appservice ase/plan`: Support ASEv3
- Fix #16026 and #16118 for az appservice plan
- Fix #16509: Add support for os-preference
- Improve behavior of appservice ase create-inbound-services to allow skipping DNS services and support DNS for ASEv2
- `az webapp up/az webapp create`: Fix nonetype errors
- `az webapp up/create`: better error handling of app name with period
- Fix #16681: `az webapp config ssl import`: Fix bug that causes failures on national clouds

ARM

- `az provider register`: Support registering management group

Backup

- Add CRR functionality for IaaSVM and other CRR commands
- `az backup protectable-item list`: Add protectable-item-type as an optional argument

BotService

- `az bot create/update`: Add Encryption features `--cmk-key-url` and `--encryption-off`
- `az bot update`: Rename Encryption-OFF arg to CMK-OFF and updating api version

Compute

- [BREAKING CHANGE] `vmss create`: Rename orchestration mode values

- New command group `sshkey`. Allow referencing a SSH key resource when creating a VM
- `az disk create/update`: Add parameter `--enable-bursting` to support disk bursting

Extension

- Support extension command prefix match for dynamic install

HDInsight

- `az hdinsight create`: Add a new parameter `--enable-compute-isolation` to support create cluster with compute isolation feature.

Key Vault

- `az keyvault key import`: Support `--curve` parameter for importing BYOK keys
- `az keyvault certificate download`: Fix deprecated/removed method call
- `az keyvault create/update`: Remove preview tag for `--enable-rbac-authorization`

Monitor

- `az monitor metrics alert create`: Fix 'resource is not found' error

NetAppFiles

- `az netappfiles account ad add`: Add parameter `--security-operators`.
- `az netappfiles volume create`: Add parameter `--smb-continuously-available`.
- `az netappfiles volume create`: Add parameter `--smb-encryption`.
- `az netappfiles`: No longer in preview mode.

Network

- [BREAKING CHANGE] `az network vrouter`: Deprecate this command group, please use `az network routeserver`.
- `az network routeserver`: Add new command group.
- `az network application-gateway create`: Add parameter `--ssl-profile-id`
- `az network application-gateway client-cert`: Manage trusted client certificate of application gateway

- `az network application-gateway ssl-profile`: Manage ssl profiles of application gateway
- Add support for private endpoint connections to DigitalTwins

Profile

- `az login`: Launch browser in WSL 2

RDBMS

- `az mysql flexible-server create --iops`: Allow user to choose IOPS for their SKU.
- Update Postgres restore command to support available zone

Search

- Upgrade to use the latest (8.0.0) azure-mgmt-search python sdk
- `az search create`: Add support for search service creation with IP rules, public endpoint access and/or msi
- `az search update`: Add support for search service update with IP rules, public endpoint access and/or msi
- `az search private-endpoint-connection`: Manage private endpoint connection to a search service
- `az search shared-private-link-resource`: Manage shared private link resources in a search service
- `az search private-link-resource`: List available private link resources in a search service

Security

- Add new commands for `az security`

SQL

- Add managed hsm regex match to SQL
- Upgrade azure-mgmt-sql to 0.26.0
- `az sql mi create/update`: Add support for maintenance configuration in managed instance operations
- Support SQL server DevOps audit policy commands

Storage

- Fix #16079: public blob gives error
- GA Storage routing reference
- Fix #9158: Cannot generate a working SAS key from a policy
- Fix #16489: Upgrade azcopy to 10.8.0
- `az storage account blob-service-properties`: Support default service version
- Fix #16519: azcopy is given more powerful SAS than needed (has write, only needs read)

Synapse

- `az synapse workspace create` : Add parameter `--key-identifier` to support to create workspace using customer-managed key.
- `az synapse workspace key`: Add CRUD cmdlets to support to manage keys under specified synapse workspace.
- `az synapse workspace managed-identity`: Add cmdlets to support CRUD managed identity to sql access setting.
- `az synapse workspace`: Add data exfiltration protection support, add parameter `--allowed-tenant-ids`.

January 19, 2021

Version 2.18.0

ACR

- `az acr create / update`: Add `--allow-trusted-services`. This parameter determines whether trusted azure services are allowed to access network restricted registries. The default is to allow.

AKS

- `az aks check-acr`: Add new check-acr command

App Service

- Fix #13907: `az webapp config ssl import`: Change command to also import App Service Certificate

- Fix #16125: `az webapp ssh`: If using a windows client, open browser to scm link
- Fix #13291: `az webapp deployment slot swap`: The command should support preserve vnet.
- [BREAKING CHANGE] Fix regression where you can't use a runtime version with a space in the name

ARM

- `az deployment` : Add support for `--query-string`
- `az ts`: Error handling improvement for `--template-file` without `--version` prohibited

Backup

- `az backup protection backup-now`: Set default retention period to 30 days

Compute

- Fix issue of none storage_profile
- Better error handling of external tokens
- Fix a vmss reimagine issue
- `az vm/vmss extension set`: New parameter `--enable-auto-upgrade`

Container

- `az container exec`: Remove eol check to avoid closing terminal before it even started on linux

DMS

- `az dms project task create`: Added task type parameter to help distinguish if a scenario is an online migration or an offline migration.
- `az dms project task cutover`: Add new command which allows tasks with an online migration task type to cutover and end the migration.
- `az dms project create/az dms project task create`: Enable MySQL and PostgreSQL projects/tasks to be created.

IoT

- Add --tags to IoT Hub create and update

Monitor

- [BREAKING CHANGE] `az monitor log-analytics workspace data-export`: Remove deprecated `--export-all-tables` parameter and require `--tables` parameter

RDBMS

- Remove the preview tag for server key and add admin commands for Postgres and MySql

Role

- Fix #11594: `az role assignment create`: Only show supported values for `--assignee-principal-type`

Storage

- Fix #16072: Upload file with big size
- Fix #12291: `az storage blob generate-sas` does not properly encode `--full-uri`
- GA PITR and blob service properties in SRP

January 04, 2021

Version 2.17.1

RDBMS

- Hotfix: `az mysql create`: Revert incorrect parameter name 'serv_name' to 'service_name'

December 29, 2020

Version 2.17.0

ACR

- Support zone redundancy

- `az acr connected-registry`: New feature for on-prem Azure Container Registry
- `az acr scope-map update`: --add and --remove are deprecated, they are renamed to --add-repo --remove-repo
- `az acr scope-map create/update`: Add support to handle Gateway actions.
- `az acr token create`: support added for gateway actions

AKS

- Fix: add arguments removed by a previous PR
- `az aks get-credentials`: Clarify documentation for get-credentials

App Service

- Allow customer to create Python 3.9 function app
- Fix #14583: az webapp up should generate default name if name isn't provided
- Fix: Better error handling when trying to create duplicate ASP in diff location

ARM

- `az ts`: Add support for --tags
- `az ts`: Support deleting a single version
- `az provider register`: Add --accept-terms for registering RPaaS
- Fix parsing JSON files with multi-line strings

ARO

- `az aro delete`: Add RBAC validation on cluster deletion
- `az aro update`: Add RBAC validation on cluster update
- Ensure worker_profile is not None before getting the subnets from

Backup

- `az backup job list`: Solve -o table bug and added backup_management_type as command input

Batch

- Upgrade data plane to [azure batch 10.0.0 ↗](#)

- [BREAKING CHANGE] az batch job task-counts: Change the output from a JSON object returning task counts to a complex JSON object that includes task counts (`taskCounts`) as well as task slot counts (`taskSlotCounts`).

Compute

- New license type RHEL_ELS_6
- Adopt track2 SDK, `azure-mgmt-compute==18.0.0`

Container

- Fix misspelling in `az container create` CLI example text.

DataBoxEdge

- New command module: support for data-box-edge devices and management

IoT

- Update device key generation
- Update identity-enabled hub tests to fix endpoint RBAC issues

Key Vault

- `az keyvault key import`: Support `--kty` for importing BYOK keys

Monitor

- `az monitor metrics alert create`: Improve error message to give more actionable insight

Network

- `az network private-endpoint create`: Add more declaration of '--subnet' and '--private-connection-resource-id'
- Change validator of application-gateway ssl-cert create
- Migrate network to track2 SDK
- Fix bug for "az network traffic-manager profile create" when using "--routing-method MultiValue"

Profile

- Fix "missing secret or certificate in order to authenticate through a service principal"

Role

- `az ad sp create-for-rbac`: Deprecate creating Contributor role assignment by default

Security

- Add secure score commands
- Fix update alert command and support new value

SQL

- `az sql dw update`: do not accept backup-storage-redundancy argument
- `az sql db update`: update backup storage redundancy as requested from command

Storage

- Fix issue #15965: Clarify how to remove multiple legal hold tags with `az storage container legal-hold [clear|set]`
- `az storage account encryption-scope`: GA support
- Fix issue #9959: Trying to download a snapshot version of a file share fails with ResourceNotFound

Synapse

- Add new cmdlets `az synapse sql ad-admin show, create, update, delete`
- Add new cmdlet `az synapse workspace firewall-rule update`
- Add new cmdlets `az synapse sql audit-policy show, update`
- Add integration runtime related cmdlets

December 08, 2020

Version 2.16.0

ACR

- Update description for KEK param

AKS

- `az aks nodepool add/update/upgrade`: Take max surge parameter
- Add support for AGIC addon
- Change MSI cluster to default

APIM

- `az apim restore`: New command to restore a backup of an API Management service

App Service

- Fix #14857: Let users update webapp config even with access restriction
- `az functionapp create`: Accept `--runtime python` and `--runtime-version 3.9` as Azure Functions v3 parameter
- Fix #16041: az webapp config ssl create results in unknown error

ARM

- `az deployment-scripts`: Remove preview flag

Backup

- Fix #14976: CLI error improvements for ValueError and AttributeError cases
- `az backup protection undelete`: Add support for AzureWorkload protection undelete using CLI
- Fix Bad Request Error for Correct Workload Type Input

CDN

- Add preview multi-origin support.
- Add BYOC auto-rotation.

Key Vault

- `az keyvault key/secret list`: Add a parameter `--include-managed` to list managed resources

Monitor

- `az monitor metrics alert create`: Support dynamic thresholds for condition parameter
- `az monitor metrics alert update`: Support dynamic thresholds for condition parameter
- `az monitor metrics alert dimension create`: Build a metric alert rule dimension
- `az monitor metrics alert condition create`: Build a metric alert rule condition

MySQL

- Add MySQL version upgrade CLI

NetAppFiles

- `az netappfiles account ad add`: Two optional parameters added, `aes_encryption` and `ldap_signing`
- `az netappfiles account backup-policy update`: Three optional parameters added named `tags`, `type` and `id`
- `az netappfiles snapshot policy create`: An optional parameter added named `provisioning_state`

Network

- `az network network watcher configure`: Fix `NetworkWatcherCountLimitReached` error caused by case sensitivity of location value
- `az network application-gateway http-listener`: Fix bug that cannot create and update with WAF policy name
- `az network route-table`: Deprecate route table V1
- `az network cross-region-lb`: Support cross-region load balancer
- `az network express-route port generate-loa`: New command to generate and download the PDF letter of authorization for a ExpressRoutePort

Packaging

- Add Ubuntu Groovy package

RDBMS

- Add single server show-connection-string and tests for local-context commands, server creation

Role

- Add long-summary/warning for commands generating credentials

Search

- Add SKU option

Service Fabric

- Update SF app docs. only support for arm deployed resources

Synapse

- Support synapse sql dw cmdlets and update az synapse workspace create cmdlet

November 20, 2020

Version 2.15.1

Profile

- Hotfix: Fix #15961: az login: UnboundLocalError: local variable 'token_entry' referenced before assignment

November 17, 2020

Version 2.15.0

ACS

- Add v3 deprecation warnings

AKS

- Add ephemeral os functionality
- Engineering improvement: Replace addon strings with constants
- `az aks install-cli`: Support customize download url
- `az aks browse`: Point to Azure Portal Kubernetes resources view if k8s >=1.19 or kube-dashboard not enabled
- Support BYO control plane identity
- `az aks use-dev-spaces`: Indicate that dev-spaces commands are deprecated

AMS

- Change "region" to "location" in output string: `az ams account sp create`

App Config

- Fix key vault client initialization

App Service

- Fix #13646: Unable to create App Service Plan in a different resource group to App Service Environment
- Fix #11698 #15198 #14862 #15409: `az webapp/functionapp config access-restriction add`
- `az functionapp create`: Add Node 14 preview support.
- `az functionapp create`: Remove preview flag from custom handlers.
- [BREAKING CHANGE] `az functionapp update`: Migrate a functionapp from Premium to Consumption plans now requires the '--force' flag.
- `az functionapp update`: Add error message if functionapp migration involves any plans on Linux.
- `az functionapp update`: Add more descriptive error message if functionapp migration fails.

ARM

- Fix an issue where What-If shows two resource group scopes with different casing
- `az deployment`: Print out error details for deployment

Backup

- Fix #14976: KeyError fixed and help text improved

Batch

- Fix #15464: Update check for pfx file without password in batch create_certificate

Billing

- [BREAKING CHANGE] az billing invoice: Remove properties BillingPeriodsNames and DownloadUrlExpiry from the response.
- az billing invoice: Support many other scopes like BillingAccount, BillingProfile and existing subscription.
- az billing account: New commands to support display and update existing billing accounts.
- az billing balance: New commands to support display balance of a billing profile.
- az billing customer: New commands to support display customer of billing account.
- az billing policy: New commands to support display and update policy of a customer or a billing profile.
- az billing product: New commands to manage products of a billing account.
- az billing profile: New commands to manage a billing profile.
- az billing property: New commands to display and update a billing account's properties.
- az billing subscription: New commands to manage the subscriptions for a billing account.
- az billing transaction: New commands to list transaction of an invoice.
- az billing agreement: New commands to manage billing agreement.
- az billing permission: New commands to manage billing permission.
- az billing role-assignment: New commands to manage role assignment.
- az billing role-definition: New commands to display role definition.
- az billing instruction: New commands to manage instructions of billing.

Compute

- Fix update permission check issue
- Enhancement of vm list-skus table format
- vm host group create: Make --platform-fault-domain-count required and update help
- Support update vm/image version when they use cross tenant images

DPS

- Allow tags in IoT DPS create command

HDInsight

- az hdinsight create: Add two parameters `--resource-provider-connection` and `--enable-private-link` to support relay outbound and private link feature.

Key Vault

- Refine error messages for HSM `list-deleted` and `purge`
- Support selective key restore for managed HSMs

NetAppFiles

- [BREAKING CHANGE] az netappfiles pool update: Remove service-level from parameters.
- az netappfiles pool update: Add optional parameter qos-type.
- az netappfiles pool create: Add optional parameter qos-type.
- az netappfiles volume replication suspend: Add force-break-replication as optional parameter.
- Add az netappfiles volume replication re-initialize: New command is added to re-initialise replication.
- Add az netappfiles volume pool-change: New command to change the pool of a volume.
- Add az netappfiles snapshot policy: New command group with list, delete, update, show, create and volumes commands.
- Add az netappfiles account backup: New command group with show, list and delete commands
- Add az netappfiles volume backups: New command group with show, list, delete, update and create commands.
- Add az netappfiles account backup-policy: New command group with show, list, delete, update and delete commands.
- Add az netappfiles vault list: New command is added.
- az netappfiles account ad add: Add optional parameters kdc-ip, ad-name, server-root-ca-certificate and backup-operators
- az netappfiles volumes create: Add optional parameters snapshot-policy-id, backup-policy-id, backup-enabled, backup-id, policy-enforced, vault-id, kerberos-enabled, throughput-mbps, snapshot-directory-visible, security-style, kerberos5-read-only, kerberos5-read-write, kerberos5i-read-only, kerberos5i-read-write, kerberos5p-read-only, kerberos5p-read-write and has-root-access.

- `az netappfiles volume update`: Add optional parameters vault-id, backup-enabled, backup-policy-id, policy-enforced and throughput-mibps

Network

- Fix bug that can't create a Standard_v2 application-gateway without a private static IP address
- `az network dns zone import`: Raise FileOperationError instead of FileNotFoundError if zone file doesn't exist
- Fix NoneType error crash while deleting nonexisting resources of ApplicationGateway, LoadBalancer, Nic

Private DNS

- `az network private-dns zone import`: Raise FileOperationError instead of FileNotFoundError if zone file doesn't exist

Profile

- `az login`: Add back the warning that a browser is opened

Role

- `az role assignment create`: Make `--description`, `--condition`, `--condition-version` preview

Security

- `az security pricing`: Update help to reflect current API version being called

Storage

- Fix #15600: `az storage fs exists`: in case fs does not exist ResourceNotFoundError is returned
- Fix #15706: The examples for storage container create are incorrect
- `az storage blob delete-batch`: Correct typo in documentation.

November 09, 2020

Version 2.14.2

App Service

- Fix #15604, #15605: Add Dotnet5 support

November 06, 2020

Version 2.14.1

ARM

- Hotfix: Add TS multiline string support for template inputs

October 27, 2020

Version 2.14.0

AKS

- Add PPG support
- Update max standard load balancer timeout to 100 minutes

APIM

- Fix issue with creating consumption tier instance

App Config

- Fix querying key-values by comma separated labels

App Service

- Bugfix: az webapp up fails when user doesn't have write permissions to project's parent directory
- Fix #13777: Fix to remove escape chars from XML
- Fix #15441: az webapp create-remote-connection fails with AttributeError: 'Thread' object has no attribute 'isAlive'

- [BREAKING CHANGE] az webapp up: add optional params (os & runtime) and updated runtimes

ARM

- Make template deployment What-If commands GA
- [BREAKING CHANGE] Add user confirmation for az ts create
- Fix the returned data when tagging multiple resources

Backup

- `az backup policy create`: Add support for IaaSVM backup policy creation from CLI
- Increasing VM protection limit from 100 to 1000

Compute

- `sig image-definition create`: add --features
- New API version of gallery_images 2020-09-30
- `az vm update / az sig image-version update`: Support update vm/image-version even it uses a cross tenant image
- Remove validation of vm host SKUs

Cosmos DB

- `az cosmosdb create/update`: Improve error message from incorrect --locations input
- `az cosmosdb sql container create/update`: Add --analytical-storage-ttl parameter

HDInsight

- [BREAKING CHANGE] az hdinsight create: remove two parameters: --public-network-access-type and --outbound-public-network-access-type

IoT Central

- Remove preview warning since it is already GAed

Key Vault

- Invalidate `--enable-soft-delete false` while creating or updating vaults

- Make `--bypass` and `--default-action` work together with network acl parameters while creating vaults

Misc.

- Add bash-completion to Dockerfile

RDBMS

- Add List-SKUS Command, Table Transformers, Local Context for Postgres, MySQL, Mariadb Single Server
- [BREAKING CHANGE] Parameter name updates. Improvements to Management Plane for MySQL and PostgreSQL
- `az postgres|mariadb|mysql server create` : Update create experience for Postgres, MySQL and MariaDB - new fields in the output , Introduce new values for `--public` parameter in create command (all,<IP>,<IPRange>,0.0.0.0)

SignalR

- `az signalr create`: Add new option `--enable-messaging-logs` for controlling service generate messaging logs or not
- `az signalr update`: Add new option `--enable-messaging-logs` for controlling service generate messaging logs or not

SQL

- [BREAKING CHANGE] Fix response for backup storage redundancy param name and value for MI
- `az sql db audit-policy show`: extend to show database's audit policy including LA and EH data
- `az sql db audit-policy update`: extend to allow LA and EH update along with database's audit policy
- `az sql db audit-policy wait`: place the CLI in a waiting state until a condition of the database's audit policy is met.
- `az sql server audit-policy show`: extend to show servers's audit policy including LA and EH data
- `az sql server audit-policy update`: extend to allow LA and EH update along with server's audit policy

- `az sql server audit-policy wait`: place the CLI in a waiting state until a condition of the server's audit policy is met.
- Add AAD-only Support for SQL Managed Instances and Servers
- `az sql db replica create`: Add --partner-database argument

Storage

- Fix #15111: `az storage logging update` fails without optional argument
- Fix bug when using set-tier command with service principal login
- Upgrade version for file datalake to 2020-02-10
- `az storage queue list`: Track2 supported
- `az storage fs access`: Support managing ACLs recursively

Synapse

- Add pipeline, linked service, trigger, notebook, data flow and dataset related cmdlets

October 13, 2020

Version 2.13.0

ACR

- `az acr helm`: Update deprecation url
- Add logtemplate and systemtask changes for ACR Tasks

AKS

- Support virtual-node with aks create: `az aks create --enable-addons virtual-node`
- Add node image only option for CLI
- Expect kube-dashboard addon be disabled by default
- `az aks create/update`: Add LicenseType support for Windows
- Support add Spot node pool
- Honor addon names defined in Azure CLI

AMS

- Fix #14687: Mixed resource group and account name in command "az ams streaming-endpoint show"

App Config

- Fix test bug
- Support AAD auth for data operations

App Service

- `az functionapp deployment source config-zip`: Fixed an issue where config-zip could throw an exception on success on linux consumption
- Bugfix: Better error messages for webapp commands
- `az appservice domain create, show-terms`: Add ability to create app service domain
- `az functionapp create`: Removed the preview flag from Java 11 when creating a new function app
- [BREAKING CHANGE] az webapp create, az webapp up - Update available webapp runtimes

ARM

- `az ts`: Add new commands for template specs
- `az deployment` : Add support for --template-spec -s

Compute

- Fix host group creation FD count limitation
- Add new command to support upgrading extensions for VMSS
- Fix the image reference is missing issue

HDInsight

- `az hdinsight create`: add deprecate information for argument --public-network-access-type and --outbound-public-network-access-type
- `az hdinsight create`: add deprecate information for argument --public-network-access-type and --outbound-public-network-access-type
- `az hdinsight create`: add parameter --idbroker to support customer to create ESP cluster with HDInsight Id Broker

IoT Central

- Remove deprecated 'az iotcentral' command module

Key Vault

- Support `--hsm-name` for `az keyvault key encrypt/decrypt`

Lab

- Fix #14127: `__init__()` takes 1 positional argument but 2 were given

Network

- `az network application-gateway ssl-cert show`: Add example to demonstrate certificate format and fetch information
- `az network application-gateway rule`: Support `--priority` option
- `az network application-gateway create`: Fix bug that cannot create without public IP specified
- `az network application-gateway waf-policy managed-rule rule-set add`: Expose server error to user to give more intuitive hint message.
- `az network application-gateway waf-policy managed-rule rule-set update`: Support to change rule set type version.

RDBMS

- Bugfix: `az postgres flexible-server create` Remove hardcoded API version from network client.

Role

- Fix #15278: `az role assignment list/delete`: Forbid empty string arguments

SQL

- `az sql mi db log-replay`: Support for log replay service on managed database
- Ignore character casing for backup storage redundancy param value for managed instance

- [BREAKING CHANGE] az sql db create: Add --backup-storage-redundancy parameter; add warning for unspecified bsr/bsr == Geo.

SQL VM

- az sql vm show: Add configuration options to --expand flag

Storage

- [BREAKING CHANGE] az storage blob copy start: Fix format issue for --destination-if-modified-since and --destination-if-unmodified-since
- [BREAKING CHANGE] az storage blob incremental-copy start: Fix format issue for --destination-if-modified-since and --destination-if-unmodified-since
- az storage fs: Fix connection string issue
- az storage share-rm: GA release access tier
- az storage container-rm: Add a new command group to use the Microsoft.Storage resource provider for container management operations.

September 29, 2020

Version 2.12.1

RDBMS

- Hotfix: az postgres flexible-server create : Update VnetName to exclude servername and update default region for MySQL

September 22, 2020

Version 2.12.0

ACR

- Fix #14811 Add support for dockerignore override

AKS

- CLI should tolerate empty kubeconfig

- FIX #12871: az aks enable-addons: Autogenerated help example is wrong for virtual-node option
- Remove legacy aci connector actions
- Support azure policy addon in azure-cli
- Fix case sensitive issue for AKS dashboard addon
- Update mgmt-containerservice to 9.4.0 and enable 09-01 API

APIM

- Support product / productapi / namedValue entity commands && bump sdk version

App Config

- Support enabling/disabling PublicNetworkAccess for existing stores

App Service

- Add support for Premium V3 pricing tier
- Fix #12653: az webapp log config --application-logging false doesn't turn it off
- Fix #14684: access-restriction remove by ip address does not work; #13837-az webapp create - Example for different RSgroups for Plan and WebApp
- functionapp: Add support for custom handlers. Deprecated Powershell 6.2.
- functionapp: Fix issue where app setting was being incorrectly set for linux custom images

ARM

- `az deployment group/sub/mg/tenant what-if`: Show "Ignore" resource changes last

Compute

- Add new license_type in vm create/update: RHEL_BYOS, SLES_BYOS
- Upgrade disk API version to 2020-06-30
- disk create: add --logical-sector-size, --tier
- disk update: Support --disk-iops-read-only, --disk-mbps-read-only, --max-shares
- New command disk-encryption-set list-associated-resources
- vm boot-diagnostics enable: --storage becomes optional
- New command: vm boot-diagnostics get-boot-log-uris
- vm boot-diagnostics get-boot-log: support managed storage

Config

- Rename local-context to config param-persist

Cosmos DB

- Support for Migration APIs for Throughput resource for Autoscale feature in CosmosDB

Eventhub

Added Cluster commands and trusted_service_access_enabled parameter for Networkruleset

Extension

- `az extension add`: Add `--upgrade` option to update the extension if already installed
- Turn on dynamic install by default

IoT

- Enabled minimum TLS version on IoT Hub Create

IoT Central

- App delete operation is now long running operation

IoT Hub

- Deprecated 'show-connection-string' command

Key Vault

- Managed HSM public preview
- Fix the issue that `--maxresults` does not take effect while listing resources or resource versions

Kusto

- Add deprecating message

Monitor

- `az monitor log-analytics workspace linked-storage`: expose detailed error message to customers

Network

- `az network vnet subnet`: Support `--disable-private-endpoint-network-policies` and `--disable-private-link-service-network-policies`
- Fix bug while updateing flow-log when its subproperty `network_watcher_flow_analytics_configuration` is None
- API version bump to 2020-06-01
- Support `--tcp-port-behavior` while configuring a TCP configuration of a Connection Monitor V2
- Support more types and coverage level while creating Endpoint of Connection Monitor V2
- Support `--host-subnet` to create VirtualHub underneath as VirtualRouter

RDBMS

- Management Plane updates for PostgreSQL and MySQL

Role

- `az role assignment create/update`: Support `--description`, `--condition` and `--condition-version`
- `az ad app permission delete`: Support `--api-permissions` to delete specific `ResourceAccess`

Service Fabric

- Add managed cluster and node type commands

SQL

- Upgrade `azure-mgmt-sql` to 0.20.0
- Add backup storage redundancy optional parameter to MI create cmdlet

Storage

- `az storage share-rm stats`: Get the usage bytes of the data stored on the share.
- GA release storage blob PITR
- `az storage blob query`: Support Azure Storage Query Acceleration
- Support Soft Delete for file share
- `az storage copy`: Add account credentials support and deprecate `--source-local-path`, `--destination-local-path`, `--destination-account-name`
- `az storage account blob-service-properties update`: Add container delete retention policy support

Synapse

- Fixed typo in example of az synapse role assignment create and delete

August 28, 2020

Version 2.11.1

ACR

- Add Isolated Tier to Agent Pool
- Add OCI Artifact Source Context

AKS

- Fix aks cluster create issue

Cognitive Services

- [BREAKING CHANGE] Show additional legal term for certain APIs

Network

- [BREAKING CHANGE] Allow to create both public and private IP while creating an Application Gateway
- `az network list-service-tags`: add details on location parameter use to the help message

Storage

- `az storage blob list`: Support OR properties with new api version

August 25, 2020

Version 2.11.0

AKS

- Remove preview tag from Virtual Node add-on
- Add AKS CMK argument in cluster creation
- Set network profile when using basic load balancer.
- Remove max pods validation from CLI and let preflight handle it
- Fixing add-ons available in the help message in `az aks create`
- Bring in support for cluster autoscaler profile in core CLI

AppService

- `az webapp`: Add list-instances command
- `az webapp ssh`: Add --instance parameter to connect to a specific instance
- `az webapp create-remote-connection`: Add --instance parameter to connect to a specific instance
- Fix #14758: az webapp create errors when creating windows app with --runtime dotnetcore
- Fix #14701: Implement functionapp create --assign-identity
- Fix #11244: `az webapp auth update`: Add optional parameter to update client-secret-certificate-thumbprint
- `az functionapp keys`: Added commands that allow users to manage their function app keys
- `az functionapp function`: Added commands that allow users to manage their individual functions
- `az functionapp function keys`: Added commands that allow users to manage their function keys
- Fix #14788: az webapp create not getting correct webapp when names are substrings
- `az functionapp create`: Removed ability to create 2.x Functions in regions that don't support it

ARM

- `az resource list`: Extend the return data of `createdTime`, `changedTime` and `provisioningState`
- `az resource`: Add parameter `--latest-include-preview` to support using the latest api-version whether this version is preview

ARO

- CLI enhancements, including route table checking permissions

Cloud

- `az cloud register`: Fix registering clouds with a config file

Compute

- Update VM SKUs that support accelerated networking
- `az vm create`: Automatic in-guest patching
- `az image builder create`: Add `--vm-size`, `--os-disk-size`, `--vnet`, `--subnet`
- New command `az vm assess-patches`

Container

- Fix #6235: Update help text for ports parameter in container create

Datalake Store

- Fix issue #14545 for data lake join operation

EventHub

- `az eventhubs eventhub create/update`: Change documentation of `destination_name`

Extension

- Add `az extension list-versions` command to list all available versions of an extension

HDInsight

- Support creating cluster with autoscale configuration and Support managing autoscale configuration
- Support creating cluster with encryption at host

IoTCentral

- CLI documentation improvements

Monitor

- `az monitor metrics alert create`: support RG and Sub as the scope values

NetAppFiles

- [BREAKING CHANGE] `az netappfiles snapshot create`: Removed file-system-id from parameters
- [BREAKING CHANGE] `az netappfiles snapshot show`: Snapshot no longer has parameter file-system-id
- `az netappfiles account`: Model ActiveDirectory has a new parameter backup_operators
- `az netappfiles volume show`: Model dataProtection has a new parameter snapshot
- `az netappfiles volume show`: Model Volume has a new parameter snapshot_directory_visible

Network

- `az network dns export`: export FQDN for MX, PTR, NS and SRV type instead of relative path
- Support private link for managed disks
- `az network application-gateway auth-cert show`: Add example to demonstrate certificate format
- `az network private-endpoint-connection`: support app configuration

RBAC

- `az ad group create`: support specify description when creating a group
- `az role definition create`: print human readable message instead of exception when assignableScope is an empty array

- [BREAKING CHANGE] `az ad sp create-for-rbac`: change default permission of created certificate

SQL

- `az sql server audit-policy`: Add sql server auditing support

Storage

- `az storage blob copy start-batch`: Fix #6018 for --source-sas
- `az storage account or-policy`: Support storage account object replication policy
- Fix issue #14083 to upgrade azure-multiapi-storage package version for package issue and new api version support
- `az storage blob generate-sas`: add examples for --ip and refine error message
- `az storage blob list`: Fix next_marker issue

Synapse

- Add workspace, sparkpool, sqlpool related cmdlets
- Add spark job related commands based on track2 sdk
- Add accesscontrol feature related commands based on track2 sdk

Upgrade

- Add `az upgrade` command to upgrade azure cli and extensions

August 11, 2020

Version 2.10.1

App Service

- Fix #9887 webapp and functionapp, support assigning/removing user managed identity
- Fix #1382, #14055: Update error messages for az webapp create and az webapp config container set
- `az webapp up`: Fix default ASP selection logic when --plan parameter is not provided

AppConfig

- Support enabling/disabling PublicNetworkAccess during store creation

Compute

- Support associating disk and snapshot with a disk-access resource

Lab

- Fix for issue #7904 date validation bug in lab vm creation

Storage

- `az storage blob upload-batch`: Fix issue #14660 with unpositional arguments

August 04, 2020

Version 2.10.0

AKS

- `az aks update`: Change --enable-aad argument to migrate a RBAC-enabled non-AAD cluster to a AKS-managed AAD cluster
- `az aks install-cli`: Add --kubelogin-version and --kubelogin-install-location arguments to install kubelogin
- Add az aks nodepool get-upgrades command

AMS

- Fix #14021: az ams account sp is not idempotent

API

- apim api import: support API import and enhance other api level cli commands

App Service

- Fix #13035: Add validation for az webapp config access-restriction to avoid adding duplicates

AppConfig

- Default to standard sku if not specified
- [BREAKING CHANGE] Support settings with JSON content type

ARM

- `az resource tag`: Fix the bug of managedApp tagging and some related test issues
- `az deployment mg/tenant what-if`: Add support to management group and tenant level deployment What-If
- `az deployment mg/tenant create`: Add --confirm-with-what-if/-c parameter.
- `az deployment mg/tenant create`: Add --what-if-result-format/-r parameter.
- `az deployment mg/tenant create`: Add --what-if-exclude-change-types/-x parameter.
- `az tag`: az tag support for resource id parameter

Backup

- Trigger AFS container/item discovery only when needed

CDN

- Add private link fields to origin

Compute

- `az vm/vmss create`: Select a valid username for user if the default username is invalid
- `az vm update`: support cross tenant image
- `az disk-access`: Add new command group to operate disk access resource
- Support dedicated host group automatic placement
- Support ppg and spg in VMSS orchestration mode

Config

- `az config`: Add new `config` command module

Extension

- Support automatically installing an extension if the extension of a command is not installed

HDInsight

- Add 3 parameters to the command `az hdinsight create` to support private link and encryption in transit feature:

IoT Hub

- Fix #7792: IoT Hub Create is not idempotent

IoT Central

- Add parameter option list for iot central

KeyVault

- `az keyvault key encrypt/decrypt`: add parameter `--data-type` for explicitly specifying the type of original data

Monitor

- `az monitor log-analytics workspace data-export`: support event hub namespace as the destination.
- `az monitor autoscale`: support namespace and dimensions for --condition

NetAppFiles

- `az volume revert`: Add Volume Revert to revert a volume to one of its snapshots.
- [BREAKING CHANGE] Remove `az netappfiles mount-target`.
- `az volume show`: Add site to Active Directory Properties

Network

- `az application-gateway private-link add`: support to specify an existing subnet by ID

- `az network application-gateway waf-policy create`: support version and type

Storage

- Fix #10302: Support guess content-type when synchronizing files
- `az storage blob lease`: Apply new api version for blob lease operations
- `az storage fs access`: Support AAD credential in managing access control for ADLS Gen2 account
- `az storage share-rm create/update`: add --access-tier to support access tier

July 16, 2020

Version 2.9.1

AKS

- Remove explicit setting of VMSS in Windows example command since it is now default

IoT

- [BREAKING CHANGE] `az iot pnp`: Remove IoT PNP preview commands from core CLI

REST

- Fix #14152: `az rest`: Accept ARM URLs without subscription ID

Storage

- Fix #14138: Make some permissions optional

July 14, 2020

Version 2.9.0

ACR

- Handle log artifact link from Registry to stream logs

- Deprecate helm2 commands

AKS

- `az aks create`: add --enable-aad argument
- `az aks update`: add --enable-aad argument

APIM

- Added general az apim api commands

AppConfig

- Add example for using --fields in appconfig revision

AppService

- `az functionapp create`: Added support for Java 11 and Powershell 7. Added Stacks API Support.
- Fix #14208 multi-container app creation fails
- Fix az webapp create - use hardcoded runtime stacks

ARM

- `az resource tag`: Fix the problem of tagging resources with resource type `Microsoft.ContainerInstance/containerGroups`

Compute

- Bump version disks 2020-05-01, compute 2020-06-01
- Double encryption of disk encryption set
- `az vmss update`: support specify cross tenant image.
- `az sig image-version create`: support specify cross tenant image.
- `vm/vmss create`: Encryption of cache & data-in-transit for OS/Data disks and temp disks for VM & VMSS
- Add simulate-eviction operation for VM and VMSS

CosmosDB

- Recent features: Autoscale, IpRules, EnableFreeTier and EnableAnalyticalStorage

EventGrid

- Add CLI support for 2020-04-01-preview and mark preview features with is_Preview=True

Find

- Fix #14094 az find Fix Queries failing when not logged in and when telemetry is disabled

HDInsight

- Add two commands to support hdinsight node reboot feature

Monitor

- Remove preview flag for commands under Log Analytics workspace
- `az monitor diagnostic-settings subscription`: Support diagnositc settings for subscription
- `az monitor metrics`: support ',' and '|' in metric name
- `az monitor log-analytics workspace data-export`: support log analytics data export

Network

- `az network application-gateway frontend-ip update`: Deprecating the --public-ip-address parameter
- Bump azure-mgmt-network to 11.0.0
- `az network express-route gateway connection`: support routing configuration
- `az network virtual-appliance`: Support Azure network virtual appliance.
- Application Gateway support private link feature

PolicyInsights

- `az policy state`: add trigger-scan command to trigger policy compliance evaluations
- `az policy state list`: expose versions of policy entities in each compliance record

Profile

- `az account get-access-token`: Show expiresOn for Managed Identity

RDBMS

- Support Minimum TLS version
- Add Infrastructure Encryption for Azure Postgres and MySQL

Security

- Add allowed_connections commands
- Add Adaptive network hardeningss commands
- Add adaptive_application_controls commands
- Addition of az security iot-solution/ iot-alerts/iot-recommendations/iot-analytics REST to Azure CLI
- Add regulatory compliance CLI

SignalR

- Add features including managing private endpoint connections, network rules and upstream

SQL

- `az sql mi create, az sql mi update`: Add `--tags` parameter to support resource tagging
- `az sql mi failover`: Support failover from primary or secondary point

Storage

- `az storage account create/update`: Add `--allow-blob-public-access` to allow or disallow public access for blob and containers
- `az storage account create/update`: Add `--min-tls-version` to support setting the minimum TLS version to be permitted on requests to storage.
- Remove check in token credential
- Fix the storage account name in examples

Webapp

- Bugfix: az webapp log deployment show - return deployment logs instead of log metadata
- Bugfix: az webapp vnet-integration add - fix error handling if bad vnet name, support vnet resource ID

June 23, 2020

Version 2.8.0

ACR

- Add support for region endpoint disable / routing disable
- [BREAKING CHANGE] `az acr login --expose-token` does not accept username and password

ACS

- Remove private cluster and 2019-10-27-preview API

AKS

- Support `--yes` for `az aks upgrade`
- Revert "change default vm sku to Standard_D2s_v3 (#13541)"
- Add "`az aks update --uptime-sla`"
- Fix typo in `az aks update` command
- Change to support 0 node agent pool and block manual scale for CAS enabled pool
- Fix typo on `VirtualMachineScaleSets` and update references to Kubernetes versions

AMS

- CHANGE help text for `--expiry` parameter.

AppService

- `az webapp log deployment show`: Show the latest deployment log, or the deployment logs of a specific deployment if deployment-id is specified
- `az webapp log deployment list`: List of deployment logs available
- Fix: Surface error when invalid webapp name provided

- Fix #13261 az webapp list-runtimes use static list until new Available Stacks API is available
- `az appservice ase create`: Fix create issue #13361
- `az appservice ase list-addresses`: Fix change of SDK #13140.
- Fix webapp/slot creation for Windows Containers
- `az webapp auth update`: Add optional parameter to update runtime-version
- Support list, delete, approve and reject private endpoint connection for webapp in CLI
- Fix #13888 : Add support for Static WebApps: get, list, create commands
- Improved error messages for SSH Tunnel Connection

ARM

- `az tag`: Add examples for -h
- `az deployment group/sub what-if`: Add --exclude-change-types/-x parameter.
- `az deployment group/sub/mg/tenant create`: Add --what-if-exclude-change-types/-x parameter.
- `az deployment group/sub/mg/tenant validate`: Show error messages in a better format.
- `az group export`: Add new parameters `--skip-resource-name-params` and `--skip-all-params` to support skip parameterization
- Add az feature unregister api

ARO

- Add Public, Private to params for help with ingress/apiserver visibility

Batch

- `az batch account create`: Add new parameter `--public-network-access`
- `az batch account create`: Add new parameter `--identity-type`
- `az batch account set`: Add new parameter `--identity-type`
- [BREAKING CHANGE] `az batch pool create`: When creating a pool using a custom image, the --image property of can now only refer to a Shared Image Gallery image.
- [BREAKING CHANGE] `az batch pool create`: When creating a pool with --json-file option and specifying a networkConfiguration, the publicIPs property has moved in to a new property publicIPAddressConfiguration. This new property also supports a new ipAddressProvisioningType property which specifies how the pool

should allocate IP's and a publicIPs property which allows for configuration of a list of PublicIP resources to use in the case ipAddressProvisioningType is set to UserManaged

- `az network private-link-resource`: Add support for the Microsoft.Batch batchAccount resource
- `az network private-endpoint-connection`: Add support for the Microsoft.Batch batchAccount resource

CDN

- `az cdn custom-domain enable-https`: Add BYOC support.
- `az cdn custom-domain enable-https`: Fix enabling custom HTTPS with CDN managed certificates for Standard_Verizon and Standard_Microsoft SKUs.

Cognitive Services

- [BREAKING CHANGE] `az cognitiveservices account` now have a unified structure for all commands.
- `az cognitiveservices account identity`: Add identity management for Cognitive Services.

Compute

- `az image builder`: Upgrade API version to 2020-02-14
- `az image builder create`: Add `--identity` to support identity configuration
- `az image builder customizer add`: Support Windows update customizer
- New command `az image builder cancel`
- Show a warning when a user deploys a VMSS pinned to a specific image version rather than latest

Cosmos DB

- `az cosmosdb`: Add exists command to database and container groups
- Allow creating fixed collections

EventHub

- `az eventhubs namespace create` : Add managed identity parameters

Extension

- Add --version to support to install from a specific version
- Enable CLI extensions to include packages in the 'azure' namespace

IoT Hub

- [BREAKING CHANGE] az iot hub job: Remove deprecated job commands

KeyVault

- `az keyvault key import`: Supports importing from strings via two new parameters.
- Support string/bytes encryption and decryption with stored keys

Monitor

- Support no wait for cluster creation
- `az monitor log-analytics workspace saved-search`: Support new commands for saved search

Network

- `az network application-gateway address-pool update`: Refine help message and add examples.
- `az network vnet create`: Support --nsg argument
- `az network lb address-pool`: Support create lb backend pool with backend address.
- `az network application-gateway address-pool`: Fix for --add argument

RBAC

- `az ad sp create-for-rbac`: Support name with space, slash and back slash
- `az ad sp create-for-rbac`: Refine error message when user specify an invalid scope

Security

- Add security assessment commands

SQL

- `az sql db ltr-policy/ltr-backup`: update/show long term retention policy, show/delete long term retention backups, restore long term retention backup

Storage

- Fix authentication issue to support get token for --subscription
- `az storage remove`: Fix issue #13459 to raise exception for operation failure
- Fix issues #13012, #13632 and #13657 to remove unused arguments for generate-sas related commands
- `az storage logging update`: Add check for logging version
- `az storage blob show`: Add more properties for blob with track 2 SDK
- Fix #13708: Refine warning message for credential
- `az storage share-rm create/update`: Add NFS protocol and root squash support
- `az storage account create`: Add support for double encryption
- [BREAKING CHANGE] `az storage blob/container/file/share/table/queue generate-sas`: make --expiry and --permissions required
- `az storage blob set-tier`: Migrate to Track 2 to support setting rehydrate priority

June 02, 2020

Version 2.7.0

ACR

- Fix a typo in an error message of token creation

AKS

- Change default vm sku to Standard_D2s_v3
- Fix creating role assignment for MSI cluster plus custom subnet

AppService

- Fix #12739 az appservice list-locations returns some invalid locations

ARM

- `az deployment`: Fix issue #13159 of incorrect message of JSON after removing comments and compressing
- `az resource tag`: Fix issue #13255 of tagging resources with resource type `Microsoft.ContainerRegistry/registries/webhooks`
- Improve the examples for the resource module

ARO

- Change CLIError to correct flag for --worker-vm-disk-size-gb

EventHub

- Fix for issue #12406 Argument --capture-interval does not update the "intervalInSeconds"

HDInsight

- Change get_json_object to shell_safe_json_parse

Monitor

- `az monitor metrics alert`: refine several help messages
- `az monitor diagnostic-settings create`: support --export-to-resource-specific argument
- Support LA workspace recover

Network

- `az network dns zone`: support - character
- `az network vpn-connection ipsec-policy`: change the --sa-lifetime and --sa-max-size to larger values in example
- Bump network to 2020-04-01
- `az network private-endpoint-connection`: support event grid
- `az network express-route list-route-tables`: fix bug that cannot list routes as table

Packaging

- Add Ubuntu Focal Package

RBAC

- `az ad sp credential reset`: modify credential generation to avoid troublesome special characters

Redis

- Fix #13529: Change documentation of parameter `enable_non_ssl_port`

Storage

- `az storage copy`: Add parameter `--follow-symlinks` to support symlinks
- Enable local context for storage account
- `az storage logging`: Fix issue #11969 to refine error message

May 19, 2020

Version 2.6.0

ACR

- Add default timeout of 5 minutes for any requests to ACR
- Support disable public network access
- `az acr token create`: expose `--days` argument
- `az acr import`: accept `--source` argument values which contain login in server name through client end correction

ACS

- Bug fix: remove fields cleanup for fields that no longer exist

AKS

- Update uptime-sla command help context
- Remove range check for updating min count for autoscaler
- Fix that cli doe not fail when user only specifies Windows password

AMS

- `az ams transform create`: Add ability to create a transform with a FaceDetector preset
- `az ams content-key-policy create` : Add ability to create a FairPlay content key policy with an offline rental configuration

AppConfig

- Bug fix for list key values with fields

AppService

- `az functionapp create`: AzureWebJobsDashboard will only be set if AppInsights is disabled
- Fix #10664- VNet Integration - Location Check Issue & fix #13257- az webapp up failing when RG needs to be created
- `az webapp|functionapp config ssl import`: Lookup key vault across resources groups in subscription and improve help and examples.
- Onboard local context for app service

ARM

- `az deployment`: Fix the problem that the templateLink will not be returned when deploying or validating template-uri
- `az deployment`: Fix the problem that deployment/validate does not support specially encoded character
- `az deployment sub/group what-if`: Fix array alignment and error handling
- `az deployment operation`: Modify the deprecate information

ARO

- Add examples to az aro create, list, list-credentials, show, delete
- Add generate_random_id function

Backup

- Allow FriendlyName in enable protection for AzureFileShare command
- Fix in IaaSVM restore-disks Command
- Add "MAB" BackupManagementType to item list command
- Add support for retrying policy update for failed items.

- Add Resume Protection functionality for Azure Virtual Machine
- Add support to specify ResourceGroup for storing instantRP during Create or Modify Policy

CI

- Support flake8 3.8.0

Compute

- New command az vm auto-shutdown
- `az vm list-skus`: Update --zone behavior, return all type skus now

Core

- Update local context on/off status to global user level

Extension

- `az extension add`: Add --system to enable installing extensions in a system path
- Support .egg-info to store wheel type extension metadata

IoT

- `az iot`: Update the IoT command module first run extension awareness message to the accurate, non-deprecated modern Id `azure-iot`.

IoT Hub

- Support for 2020-03-01 API and Network Isolation commands

NetAppFiles

- `az volume create`: Adds snapshot-id as a parameter to create volume this will allow users to create a volume from existing snapshot.

Network

- Fix ttl value changed unintended for dns add-record

- `az network public-ip create`: Inform customers of a coming breaking change
- Support generic commands for private link scenario
- `az network private-endpoint-connection`: Support mysql, postgres and mariadb types
- `az network private-endpoint-connection`: Support cosmosdb types
- `az network private-endpoint`: deprecate --group-ids and redirect to --group-id

Output

- Show update instruction in find, feedback and --help

Packaging

- Build MSI/Homebrew packages with dependecies resolved from requirements.txt

RBAC

- `az ad sp credential reset`: fix weak credential generation

Storage

- `az storage account file-service-properties update/show`: Add File Properties Support for Storage Account
- `az storage container create`: Fix #13373 by adding validator for public access
- Add ADLS Gen2 track2 support
- `az storage blob sync`: Support `--connection-string`
- `az storage blob sync`: Fix the incorrect error message when azcopy cannot find the installation location

April 30, 2020

Version 2.5.1

ACR

- `az acr check-health`: Fix "DOCKER_PULL_ERROR" on Windows

Compute

- `az vm list-ip-addresses`: Error handling
- Fix a bug of vm create if endpoint_vm_image_alias_doc is not set in cloud profile
- `az vmss create`: Add --os-disk-size-gb

Cosmos DB

- `az cosmosdb create/update`: add --enable-public-network support

Extension

- Fix loading wrong metadata for wheel type extension

Packaging

- Add az script for Git Bash/Cygwin on Windows

SQL

- `az sql instance-pool`: Add instance pools command group

Storage

- Upgrade package azure-multiapi-storage to 0.3.0
- Support GZRS for storage account creation and update
- `az storage account failover`: Add support for grs/gzrs storage account failover
- `az storage blob upload`: Add --encryption-scope parameter to support specifying encryption scope information

April 28, 2020

Version 2.5.0

ACS

- [BREAKING CHANGE] az openshift create: remove --vnet-peer parameter.
- `az openshift create`: add flags to support private cluster.
- `az openshift`: upgrade to `2019-10-27-preview` API version.
- `az openshift`: add `update` command.

AKS

- `az aks create`: Add support for Windows

AppService

- `az webapp deployment source config-zip`: remove sleep after request.get()

ARM

- Add template deployment What-If commands

ARO

- `az aro`: Fix table output

CI

- Onboard pytest and deprecate nose for Automation Test

Compute

- `az vmss disk detach`: fix data disk NoneType issue
- `az vm availability-set list`: Support showing VM list
- `az vm list-skus`: Fix display problem of table format

KeyVault

- Add new parameter `--enable-rbac-authorization` during creating or updating

Monitor

- Support LA cluster CMK features
- `az monitor log-analytics workspace linked-storage`: supports BYOS features

Network

- `az network security-partner`: support security partner provider

Privatedns

- Add feature in private DNS zone to import export zone file

April 21, 2020

Version 2.4.0

ACR

- `az acr run --cmd`: disable working directory override
- Support dedicated data endpoint

AKS

- `az aks list -o table` should show privateFqdn as fqdn for private clusters
- Add --uptime-sla
- Update containerservice package
- Add node public IP support
- Fix typo in the help command

AppConfig

- Resolve key vault reference for kv list and export commands
- Bug fix for list key values

AppService

- `az functionapp create`: Changed the way linuxFxVersion was being set for dotnet linux function apps. This should fix a bug that was preventing dotnet linux consumption apps from being created
- [BREAKING CHANGE] `az webapp create`: fix to keep existing AppSettings with az webapp create
- [BREAKING CHANGE] `az webapp up`: fix to create RG for az webapp up command when using -g flag
- [BREAKING CHANGE] `az webapp config`: fix to show values for non-JSON output with az webapp config connection-string list

ARM

- `az deployment create/validate`: Add parameter `--no-prompt` to support skipping the prompt of missing parameters for ARM template
- `az deployment group/mg/sub/tenant validate`: Support comments in deployment parameter file
- `az deployment`: Remove `is_preview` for parameter `--handle-extended-json-format`
- `az deployment group/mg/sub/tenant cancel`: Support cancel deployment for ARM template
- `az deployment group/mg/sub/tenant validate`: Improve the error message when deployment verification fails
- `az deployment-scripts`: Add new commands for DeploymentScripts
- `az resource tag`: Add parameter `--is-incremental` to support adding tags to resource incrementally

ARO

- `az aro`: Add Azure RedHat OpenShift V4 aro command module

Batch

- Update Batch API

Compute

- `az sig image-version create`: Add storage account type Premium_LRS
- `az vmss update`: Fix terminate notification update issue
- `az vm/vmss create`: Add support for specialized image version
- SIG API Version 2019-12-01
- `az sig image-version create`: Add `--target-region-encryption`
- Fix tests fail when running in serial due to keyvault name is duplicated in global in-memory cache

CosmosDB

- Support `az cosmosdb private-link-resource/private-endpoint-connection`

IoT Central

- Deprecate `az iotcentral`
- Add `az iot central` command module

Monitor

- Support private link scenario for monitor
- Fix wrong mocking way in test_monitor_general_operations.py

Network

- Deprecate sku for public ip update command
- `az network private-endpoint`: Support private dns zone group
- Enable local context feature for vnet/subnet parameter
- Fix wrong usage example in test_nw_flow_log_delete

Packaging

- Drop support for Ubuntu/Disco package

RBAC

- `az ad app create/update`: support --optional-claims as a parameter

RDBMS

- Add Azure active directory administrator commands for PostgreSQL and MySQL

Service Fabric

- Fix #12891: `az sf application update --application-parameters` removes old parameters that are not in the request
- Fix #12470 az sf create cluster, fix bugs in update durability and reliability and find vmss correctly through the code given a node type name

SQL

- Add `az sql mi op list`, `az sql mi op get`, `az sql mi op cancel`
- `az sql midb`: update/show long term retention policy, show/delete long term retention backups, restore long term retention backup

Storage

- Upgrade azure-mgmt-storage to 9.0.0

- `az storage logging off`: Support turning off logging for a storage account
- `az storage account update`: Enable key auto-rotated for CMK
- `az storage account encryption-scope create/update/list/show`: Add support to customize encryption scope
- `az storage container create`: Add --default-encryption-scope and --deny-encryption-scope-override to set encryption scope for container level

Survey

- Add switch to turn off survey link

April 01, 2020

Version 2.3.1

ACR

- Fix wrong version of `azure-mgmt-containerregistry` for Linux

Profile

- `az login`: Fix login failure with cloud profiles other than `latest`

March 31, 2020

Version 2.3.0

ACR

- '`az acr task update- az acr import: Modify help and error message to clarify the usage of --source and --registry
- Add a validator for argument 'registry_name'
- az acr login: Remove the preview flag on '--expose-token'
- [BREAKING CHANGE] 'az acr task create/update' Branch parameter is removed
- 'az acr task update' Customer now can update context, git-token, and or triggers individually
- 'az acr agentpool`

AKS

- Fix apiServerAccessProfile when updating --api-server-authorized-ip-ranges
- aks update: Override outbound IPs with input values when update
- Do not create SPN for MSI clusters and support attach acr to MSI clusters

AMS

- Fix #12469: adding Fairplay content-key-policy fails due to problems with 'ask' parameter

AppConfig

- Add --skip-keyvault for kv export

AppService

- Fix #12509: Remove the tag to az webapp up by default
- az functionapp create: Updated --runtime-version help menu and added warning when user specifies --runtime-version for dotnet
- az functionapp create: Updated the way javaVersion was being set for Windows function apps

ARM

- az deployment create/validate: Use --handle-extended-json-format by default
- az lock create: Add examples of creating subresource in the help documentation
- az deployment {group/mg/sub/tenant} list: Support provisioningState filtering
- az deployment: Fix the parse bug for comment under the last argument

Backup

- Added multiple files restore capabilities
- Added support for Backing up OS Disks only
- Added restore-as-unmanaged-disk parameter to specify unmanaged restore

Compute

- az vm create: Add NONE option of --nsg-rule
- az vmss create/update: remove vmss automatic repairs preview tag

- az vm update: Support --workspace
- Fix a bug in VirtualMachineScaleSetExtension initialization code
- Upgrade VMAccessAgent version to 2.4
- az vmss set-orchestration-service-state: support vmss set orchestration service state
- Upgrade disk API version to 2019-11-01
- az disk create: add --disk-iops-read-only, --disk-mbps-read-only, --max-shares, --image-reference, --image-reference-lun, --gallery-image-reference, --gallery-image-reference-lun

Cosmos DB

- Fix missing --type option for deprecation redirections

Docker

- Update to Alpine 3.11 and Python 3.6.10

Extension

- Allow to load extensions in the system path via packages

HDInsight

- (az hdinsight create:) Support customers specify minimal supported tls version by using parameter `--minimal-tls-version`. The allowed value is 1.0,1.1,1.2

IoT

- Add codeowner
- az iot hub create : Change default sku to S1 from F1
- iot hub: Support IoTHub in the profile of 2019-03-01-hybrid

IoTCentral

- Update error details, update default application template and prompt message

KeyVault

- Support certificate backup/restore

- keyvault create/update: Support --retention-days
- No longer display managed keys/secrets while listing
- az keyvault create: support `--network-acls`, `--network-acls-ips` and `--network-accls-vnets` for specifying network rules while creating vault

Lock

- az lock delete fix bug: az lock delete does not work on Microsoft.DocumentDB

Monitor

- az monitor clone: support clone metric rules from one resource to another
- Fix IcM179210086: unable to create custom metric alert for their Application Insights metric

NetAppFiles

- az volume create: Allow data protection volumes adding replication operations: approve, suspend, resume, status, remove

Network

- az network application-gateway waf-policy managed-rule rule-set add: support Microsoft_BotManagerRuleSet
- network watcher flow-log show: fix wrong deprecating info
- support host names in application gateway listener
- az network nat gateway: support create empty resource without public ip or public ip prefix
- Support vpn gateway generation
- Support `--if-none-match` in `az network dns record-set {} add-record`

Packaging

- Drop support for python 3.5

Profile

- az login: Show warning for MFA error

RDBMS

- Add server data encryption key management commands for PostgreSQL and MySQL

March 10, 2020

Version 2.2.0

ACR

- Fix: `az acr login` wrongly raise error
- Add new command `az acr helm install-cli`
- Add private link and CMK support
- add 'private-link-resource list' command

AKS

- fix the aks browse in cloud shell
- az aks: Fix monitoring addon and agentpool NoneType errors
- Add --nodepool-tags to node pool when creating azure kubernetes cluster
- Add --tags when adding or updating a nodepool to cluster
- aks create: add `--enable-private-cluster`
- add --nodepool-labels when creating azure kubernetes cluster
- add --labels when adding a new nodepool to azure kubernetes cluster
- add missing / in the dashboard url
- Support create aks clusters enabling managed identity
- az aks: Validate network plugin to be either "azure" or "kubenet"
- az aks: Add aad session key support
- [BREAKING CHANGE] az aks: support msi changes for GF and BF for omsagent (Container monitoring)(#1)
- az aks use-dev-spaces: Adding endpoint type option to the use-dev-spaces command to customize the endpoint created on an Azure Dev Spaces controller

AppConfig

- Unblock using "kv set" to add keyvault reference and feature ...

AppService

- az webapp create : Fix issue when running the command with --runtime
- az functionapp deployment source config-zip: Add an error message if resource group or function name are invalid/don't exist
- functionapp create: Fix the warning message that appears with `functionapp create` today which cites a `--functions_version` flag but erroneously uses a `_` instead of a `-` in the flag name
- az functionapp create: Updated the way linuxFxVersion and container image name were being set for linux function apps
- az functionapp deployment source config-zip: Fix an issue caused by app settings change racing condition during zip deploy, giving 5xx errors during deployment
- Fix #5720946: az webapp backup fails to set name

ARM

- az resource: Improve the examples of the resource module
- az policy assignment list: Support listing policy assignments at Management Group scope
- Add `az deployment group` and `az deployment operation group` for template deployment at resource groups. This is a duplicate of `az group deployment` and `az group deployment operation`
- Add `az deployment sub` and `az deployment operation sub` for template deployment at subscription scope. This is a duplicate of `az deployment` and `az deployment operation`
- Add `az deployment mg` and `az deployment operation mg` for template deployment at management groups
- Add `az deployment tenant` and `az deployment operation tenant` for template deployment at tenant scope
- az policy assignment create: Add a description to the `--location` parameter
- az group deployment create: Add parameter `--aux-tenants` to support cross tenants

CDN

- Add CDN WAF commands

Compute

- az sig image-version: add `--data-snapshot-luns`

- az ppg show: add --colocation-status to enable fetching the colocation status of all the resources in the proximity placement group
- az vmss create/update: support automatic repairs
- [BREAKING CHANGE] az image template: rename template to builder
- az image builder create: add --image-template

Cosmos DB

- Add Sql stored procedure, udf and trigger cmdlets
- az cosmosdb create: add --key-uri to support adding key vault encryption information

KeyVault

- keyvault create: enable soft-delete by default

Monitor

- az monitor metrics alert create: support ~ in --condition

Network

- az network application-gateway rewrite-rule create: support url configuration
- az network dns zone import: --zone-name will be case insensitive in the future
- az network private-endpoint/private-link-service: remove preview label
- az network bastion: support bastion
- az network vnet list-available-ips: support list available ips in a vnet
- az network watcher flow-log create/list/delete/update: add new commands to manage watcher flow log and exposing --location to identify watcher explicitly
- az network watcher flow-log configure: deprecated
- az network watcher flow-log show: support --location and --name to get ARM-formatted result, deprecated old formatted output

Policy

- az policy assignment create: Fix the bug that automatically generated name of policy assignment exceeds the limit

RBAC

- az ad group show: fix --group value treated as regex problem

RDBMS

- Bump the azure-mgmt-rdbms SDK version to 2.0.0
- az postgres private-endpoint-connection: manage postgres private endpoint connections
- az postgres private-link-resource: manage postgres private link resources
- az mysql private-endpoint-connection: manage mysql private endpoint connections
- az mysql private-link-resource: manage mysql private link resources
- az mariadb private-endpoint-connection: manage mariadb private endpoint connections
- az mariadb private-link-resource: manage mariadb private link resources
- Updating RDBMS Private Endpoint Tests

SQL

- Sql midb Add: list-deleted, show-deleted, update-retention, show-retention
- (sql server create:) Add optional public-network-access 'Enable'/'Disable' flag to sql server create
- (sql server update:) make some customer-facing change
- Add minimal_tls_version property for MI and SQL DB

Storage

- az storage blob delete-batch: Misbehaving `--dryrun` flag
- az storage account network-rule add (bug fix): add operation should be idempotent
- az storage account create/update: Add Routing Preference support
- Upgrade azure-mgmt-storage version to 8.0.0
- az storage container immutability create: add `--allow-protected-append-write` parameter
- az storage account private-link-resource list: Add support to list private link resources for storage account
- az storage account private-endpoint-connection approve/reject/show/delete: Support to manage private endpoint connections
- az storage account blob-service-properties update: add `--enable-restore-policy` and `--restore-days`
- az storage blob restore: Add support to restore blob ranges

February 18, 2020

Version 2.1.0

ACR

- Add a new argument `--expose-token` for `az acr login`
- Fix the incorrect output of `az acr task identity show -n Name -r Registry -o table`
- az acr login: Throw a CLIError if there are errors returned by docker command

ACS

- aks create/update: add `--vnet-subnet-id` validation

Aladdin

- Parse generated examples into commands' `_help.py`

AMS

- az ams is GA now

AppConfig

- Revise help message to exclude unsupported key/label filter
- Remove preview tag for most commands excluding managed identity and feature flags
- Add customer managed key when updating stores

AppService

- az webapp list-runtimes: Fix the bug for list-runtimes
- Add az webapp|functionapp config ssl create
- Add support for v3 function apps and node 12

ARM

- az policy assignment create: Fix the error message when the `--policy` parameter is invalid
- az group deployment create: Fix "stat: path too long for Windows" error when using large parameters.json file

Backup

- Fix for item level recovery flow in OLR
- Add restore as files support for SQL and SAP Databases

Compute

- vm/vmss/availability-set update: add `--ppg` to allowing updating ProximityPlacementGroup
- vmss create: add `--data-disk-iops` and `--data-disk-mbps`
- az vm host: remove preview tag for `vm host` and `vm host group`
- [BREAKING CHANGE] Fix #10728: `az vm create`: create subnet automatically if vnet is specified and subnet not exists
- Increase robustness of vm image list

Eventhub

- Azure Stack support for 2019-03-01-hybrid profile

KeyVault

- az keyvault key create: add a new value `import` for parameter `--ops`
- az keyvault key list-versions: support parameter `--id` for specifying keys
- Support private endpoint connections

Network

- Bump to azure-mgmt-network 9.0.0
- az network private-link-service update/create: support `--enable-proxy-protocol`
- Add connection Monitor V2 feature

Packaging

- [BREAKING CHANGE] Drop support for Python 2.7

Profile

- Preview: Add new attributes `homeTenantId` and `managedByTenants` to subscription accounts. Please re-run `az login` for the changes to take effect
- `az login`: Show a warning when a subscription is listed from more than one tenants and default to the first one. To select a specific tenant when accessing this subscription, please include `--tenant` in `az login`

Role

- `az role assignment create`: Fix the error that assigning a role to a service principal by display name yields a HTTP 400

SQL

- Update SQL Managed Instance cmdlet `az sql mi update` with two new parameters: tier and family

Storage

- [BREAKING CHANGE] `az storage account create`: Change default storage account kind to StorageV2

February 04, 2020

Version 2.0.81

ACS

- Add support to set outbound allocated ports and idle timeouts on standard load balancer
- Update to API Version 2019-11-01

ACR

- [BREAKING CHANGE] `az acr delete` will prompt
- [BREAKING CHANGE] 'az acr task delete' will prompt
- Add a new command group 'az acr taskrun show/list/delete' for taskrun management

AKS

- Each cluster gets a separate service principal to improve isolation

AppConfig

- Support import/export of keyvault references from/to appservice
- Support import/export of all labels from appconfig to appconfig
- Validate key and feature names before setting and importing
- Expose sku modification for configuration store.
- Add command group for managed identity.

AppService

- Azure Stack: surface commands under the profile of 2019-03-01-hybrid
- functionapp: Add ability to create Java function apps in Linux

ARM

- Fix issue #10246: `az resource tag` crashes when the parameter `--ids` passed in is resource group ID
- Fix issue #11658: `az group export` command does not support `--query` and `--output` parameters
- Fix issue #10279: The exit code of `az group deployment validate` is 0 when the verification fails
- Fix issue #9916: Improve the error message of the conflict between tag and other filter conditions for `az resource list` command
- Add new parameter `--managed-by` to support adding managedBy information for command `az group create`

Azure Red Hat OpenShift

- Add `monitor` subgroup to manage Log Analytics monitoring in Azure Red Hat Openshift cluster

BotService

- Fix issue #11697: `az bot create` is not idempotent
- Change name-correcting tests to run in Live-mode only

CDN

- Add support for rulesEngine feature
- Add new commands group 'cdn endpoint rule' to manage rules
- Update azure-mgmt-cdn version to 4.0.0 to use api version 2019-04-15

Deployment Manager

- Add list operation for all resources.
- Enhance step resource for new step type.
- Update azure-mgmt-deploymentmanager package to use version 0.2.0.

IoT

- Deprecate 'IoT hub Job' commands.

IoT Central

- Support app creation/update with the new sku name ST0, ST1, ST2.

Key Vault

- Add a new command `az keyvault key download` for downloading keys.

Misc

- Fix #6371: Support filename and environment variable completion in Bash

Network

- Fix #2092: az network dns record-set add/remove: add warning when record-set is not found. In the future, an extra argument will be supported to confirm this auto creation.

Policy

- Add new command `az policy metadata` to retrieve rich policy metadata resources
- `az policy remediation create`: Specify whether compliance should be re-evaluated prior to remediation with the `--resource-discovery-mode` parameter

Profile

- `az account get-access-token`: Add `--tenant` parameter to acquire token for the tenant directly, needless to specify a subscription

RBAC

- [BREAKING CHANGE] Fix #11883: `az role assignment create`: empty scope will prompt error

Security

- Add new commands `az atp show` and `az atp update` to view and manage advanced threat protection settings for storage accounts.

SQL

- `sql dw create`: deprecate `--zone-redundant` and `--read-replica-count` parameters. These parameters do not apply to DataWarehouse.
- [BREAKING CHANGE] `az sql db create`: Remove "WideWorldImportersStd" and "WideWorldImportersFull" as documented allowed values for "az sql db create --sample-name". These sample databases would always cause creation to fail.
- Add New commands `sql db classification show/list/update/delete` and `sql db classification recommendation list/enable/disable` to manage sensitivity classifications for SQL databases.
- `az sql db audit-policy`: Fix for empty audit actions and groups

Storage

- Add a new command group `az storage share-rm` to use the Microsoft.Storage resource provider for Azure file share management operations.
- Fix issue #11415: permission error for `az storage blob update`
- Integrate Azcopy 10.3.3 and support Win32.
- `az storage copy`: Add `--include-path`, `--include-pattern`, `--exclude-path` and `--exclude-pattern` parameters
- `az storage remove`: Change `--inlcude` and `--exclude` parameters to `--include-path`, `--include-pattern`, `--exclude-path` and `--exclude-pattern` parameters
- `az storage sync`: Add `--include-pattern`, `--exclude-path` and `--exclude-pattern` parameters

ServiceFabric

- Add new commands to manage application and services.

January 13, 2020

Version 2.0.80

Compute

- disk update: Add --disk-encryption-set and --encryption-type
- snapshot create/update: Add --disk-encryption-set and --encryption-type

Storage

- Upgrade azure-mgmt-storage version to 7.1.0
- az storage account create: Add --encryption-key-type-for-table and --encryption-key-type-for-queue to support Table and Queue Encryption Service

January 07, 2020

Version 2.0.79

ACR

- [BREAKING CHANGE] Remove '--os' parameter for 'acr build', 'acr task create/update', 'acr run', and 'acr pack'. Use '--platform' instead.

AppConfig

- Add support for importing/exporting feature flags
- Add new command 'az appconfig kv set-keyvault' for creating keyvault reference
- Support various naming conventions when exporting feature flags to file

AppService

- Fix issue #7154: Updating documentation for command <> to use back ticks instead of single quotes

- Fix issue #11287: webapp up: By default make the app created using up 'should be 'SSL enabled'
- Fix issue #11592: Add az webapp up flag for html static sites

ARM

- Fix `az resource tag`: Recovery Services Vault tags cannot be updated

Backup

- Added new command 'backup protection undelete' to enable soft-delete feature for IaaSVM workload
- Added new parameter '--soft-delete-feature-state' to set backup-properties command
- Added disk exclusion support for IaaSVM workload

Compute

- Fix `vm create` failure in Azure Stack profile.
- vm monitor metrics tail/list-definitions: support query metric and list definitions for a vm.
- Add new reapply command action for az vm

HDInsight

- Support for creating a Kafka cluster with Kafka Rest Proxy
- Upgrade azure-mgmt-hdinsight to 1.3.0

Misc.

- Add preview command `az version show` to show the versions of Azure CLI modules and extensions in JSON format by default or format configured by --output

Event Hubs

- [BREAKING CHANGE] Remove 'ReceiveDisabled' status option from command 'az eventhubs eventhub update' and 'az eventhubs eventhub create'. This option is not valid for Event Hub entities.

Service Bus

- [BREAKING CHANGE] Remove 'ReceiveDisabled' status option from command 'az servicebus topic create', 'az servicebus topic update', 'az servicebus queue create', and 'az servicebus queue update'. This option is not valid for Service Bus topics and queues.

RBAC

- Fix #11712: `az ad app/sp show` does not return exit code 3 when the application or service principal does not exist

Storage

- `az storage account create`: Remove preview flag for --enable-hierarchical-namespace parameter
- Update azure-mgmt-storage version to 7.0.0 to use api version 2019-06-01
- Add new parameters `--enable-delete-retention` and `--delete-retention-days` to support managing delete retention policy for storage account blob-service-properties.

December 17, 2019

2.0.78

ACR

- Added support Local context in acr task run

ACS

- [BREAKING CHANGE]az openshift create: rename `--workspace-resource-id` to `--workspace-id`.

AMS

- Updated show commands to return 3 when resource not found

AppConfig

- Fixed bug when appending api-version to request url. The existing solution doesn't work with pagination.
- Added support for showing languages besides English as our backend service support unicode for globalization.

AppService

- Fixed issue #11217: webapp: az webapp config ssl upload should support slot parameter
- Fixed issue #10965: Error: Name cannot be empty. Allow remove by ip_address and subnet
- Added support for importing certificates from Key Vault `az webapp config ssl import`

ARM

- Updated azure-mgmt-resource package to use 6.0.0
- Cross Tenant Support for `az group deployment create` command by adding new parameter `--aux-subs`
- Added new parameter `--metadata` to support adding metadata information for policy set definitions.

Backup

- Added Backup support for SQL and SAP Hana workload.

BotService

- [Breaking change] Remove '--version' flag from preview command 'az bot create'. Only v4 SDK bots are supported.
- Added name availability check for 'az bot create'.
- Added support for updating the icon URL for a bot via 'az bot update'.
- Added support for updating a Direct Line channel via 'az bot directline update'.
- Added '--enable-enhanced-auth' flag support to 'az bot directline create'.
- The following command groups are GA and not in preview: 'az bot authsetting'.
- The following commands in 'az bot' are GA and not in preview: 'create', 'prepare-deploy', 'show', 'delete', 'update'.
- Fixed 'az bot prepare-deploy' changing '--proj-file-path' value to lower case (e.g. "Test.csproj" to "test.csproj").

Compute

- vmss create/update: Added --scale-in-policy, which decides which virtual machines are chosen for removal when a VMSS is scaled-in.
- vm/vmss update: Added --priority.
- vm/vmss update: Added --max-price.
- Added disk-encryption-set command group (create, show, update, delete, list).
- disk create: Added --encryption-type and --disk-encryption-set.
- vm/vmss create: Added --os-disk-encryption-set and --data-disk-encryption-sets.

Core

- Removed support for Python 3.4
- Plug in HaTS survey in multiple commands

DLS

- Updated ADLS sdk version (0.0.48).

Install

- Install script support python 3.8

IOT

- [BREAKING CHANGE] Removed --failover-region parameter from manual-failover. Now it will failover to assigned geo-paired secondary region.

Key Vault

- Fixed #8095: `az keyvault storage remove`: improve the help message
- Fixed #8921: `az keyvault key/secret/certificate list/list-deleted/list-versions`: fix the validation bug on parameter `--maxresults`
- Fixed #10512: `az keyvault set-policy`: improve the error message when none of `-object-id`, `--spn` or `--upn` is specified
- Fixed #10846: `az keyvault secret show-deleted`: when `--id` is specified, `--name/-n` is not required
- Fixed #11084: `az keyvault secret download`: improve the help message of parameter `--encoding`

Network

- az network application-gateway probe: Added support --port option to specify a port for probing backend servers when create and update
- az network application-gateway url-path-map create/update: bug fix for `--waf-policy`
- az network application-gateway: Added support `--rewrite-rule-set`
- az network list-service-aliases: Added support list service aliases which can be used for Service Endpoint Policies
- az network dns zone import: Added support .@ in record name

Packaging

- Added back edge builds for pip install
- Added Ubuntu eoan package

Policy

- Added support for Policy API version 2019-09-01.
- az policy set-definition: Added support grouping within policy set definitions with `--definition-groups` parameter

Redis

- Added preview param `--replicas-per-master` to `az redis create` command
- Updated azure-mgmt-redis from 6.0.0 to 7.0.0rc1

ServiceFabric

- Fixed in node-type add logic including #10963: Adding new node type with durability level Gold will always throw CLI error
- Updated ServiceFabricNodeVmExt version to 1.1 in creation template

SQL

- Added "--read-scale" and "--read-replicas" parameters to sql db create and update commands, to support read scale management.

Storage

- GA Release Large File Shares property for storage account create and update command
- GA Release User Delegation SAS token Support
- Added new commands `az storage account blob-service-properties show` and `az storage account blob-service-properties update --enable-change-feed` to manage blob service properties for storage account.
- [COMING BREAKING CHANGE] `az storage copy`: `*` character is no longer supported as a wildcard in URL, but new parameters `--include-pattern` and `--exclude-pattern` will be added with `*` wildcard support.
- Fixed issue #11043: Added support to remove whole container/share in `az storage remove` command

November 26, 2019

Version 2.0.77

ACR

- Deprecated parameter `--branch` from acr task create/update

Azure Red Hat OpenShift

- Added `--workspace-resource-id` flag to allow creation of Azure Red Hat Openshift cluster with monitoring
- Added `monitor_profile` to create Azure Red Hat OpenShift cluster with monitoring

AKS

- Added support cluster certificate rotation operation using "az aks rotate-certs".

AppConfig

- Added support for using ":" for `as az appconfig kv import` separator
- Fixed issue for listing key values with multiple labels including null label.
- Updated management plane sdk, `azure-mgmt-appconfiguration`, to version 0.3.0.

AppService

- Fixed issue #11100: AttributeError for `az webapp up` when create service plan

- az webapp up: Forcing the creation or deployment to a site for supported languages, no defaults used.
- Added support for App Service Environment: az appservice ase show | list | list-addresses | list-plans | create | update | delete

Backup

- Fixed issue in az backup policy list-associated-items. Added optional BackupManagementType parameter.

Compute

- Upgraded API version of compute, disks, snapshots to 2019-07-01
- vmss create: Improvement for --orchestration-mode
- sig image-definition create: Added --os-state to allow specifying whether the virtual machines created under this image are 'Generalized' or 'Specialized'
- sig image-definition create: Added --hyper-v-generation to allow specifying the hypervisor generation
- sig image-version create: Added support --os-snapshot and --data-snapshots
- image create: Added --data-disk-caching to allow specifying caching setting of data disks
- Upgraded Python Compute SDK to 10.0.0
- vm/vmss create: Added 'Spot' to 'Priority' enum property
- [Breaking change] Renamed '--max-billing' parameter to '--max-price', for both VM and VMSS, to be consistent with Swagger and Powershell cmdlets
- vm monitor log show: Added support for querying log over linked log analytics workspace.

IOT

- Fix #2531: Added convenience arguments for hub update.
- Fix #8323: Added missing parameters to create storage custom endpoint.
- Fix regression bug: Reverted the changes which overrides the default storage endpoint.

Key Vault

- Fixed #11121: When using `az keyvault certificate list`, passing `--include-pending` now doesn't require a value of `true` or `false`

NetAppFiles

- Upgraded azure-mgmt-netapp to 0.7.0 which includes some additional volume properties associated with upcoming replication operations

Network

- application-gateway waf-config: deprecated
- application-gateway waf-policy: Added subgroup managed-rules to manage managed rule sets and exclusion rules
- application-gateway waf-policy: Added subgroup policy-setting to manage global configuration of a waf-policy
- [BREAKING CHANGE] application-gateway waf-policy: Renamed subgroup rule to custom-rule
- application-gateway http-listener: Added --firewall-policy when create
- application-gateway url-path-map rule: Added --firewall-policy when create

Packaging

- Rewrote the az wrapper in Python
- Added support for Python 3.8
- Changed to Python 3 for RPM package

Profile

- Polished error when running `az login -u {} -p {}` with Microsoft account
- Polished `SSLError` when running `az login` behind a proxy with self-signed root certificate
- Fixed #10578: `az login` hangs when more than one instances are launched at the same time on Windows or WSL
- Fixed #11059: `az login --allow-no-subscriptions` fails if there are subscriptions in the tenant
- Fixed #11238: After renaming a subscription, logging in with MSI will result in the same subscription appearing twice

RBAC

- Fixed #10996: Polish error for `--force-change-password-next-login` in `az ad user update` when `--password` is not specified

Redis

- Fixed #2902: Avoid setting memory configs while updating Basic SKU cache

Reservations

- Upgraded SDK Version to 0.6.0
- Added billingplan details info after calling Get-Gatalogs
- Added new command `az reservations reservation-order calculate` to calculate the price for a reservation
- Added new command `az reservations reservation-order purchase` to purchase a new reservation

Rest

- Changed `az rest` to GA

SQL

- Updated `azure-mgmt-sql` to version 0.15.0.

Storage

- `storage account create`: Added `--enable-hierarchical-namespace` to support filesystem semantics in blob service.
- Removed unrelated exception from error message
- Fixed issues with incorrect error message "You do not have the required permissions needed to perform this operation." when blocked by network rules or `AuthenticationFailed`.

November 4, 2019

Version 2.0.76

ACR

- Added a preview parameter `--pack-image-tag` to command `az acr pack build`.
- Added support for enabling auditing on creating a registry
- Added support for Repository-scoped RBAC

AKS

- Added `--enable-cluster-autoscaler`, `--min-count` and `--max-count` to the `az aks create` command, which enables cluster autoscaler for the node pool.
- Added the above flags as well as `--update-cluster-autoscaler` and `--disable-cluster-autoscaler` to the `az aks update` command, allowing updates to cluster autoscaler.

AppConfig

- Added appconfig feature command group to manage feature flags stored in an App Configuration.
- Fixed minor bug for appconfig kv export to file command. Stop reading dest file contents during export.

AppService

- `az appservice plan create`: Added support to set 'persistscaling' on appservice plan create.
- Fixed an issue where webapp config ssl bind operation was removing existing tags from the resource
- Added `--build-remote` flag for `az functionapp deployment source config-zip` to support remote build action during function app deployment.
- Changed default node version on function apps to ~10 for Windows
- Added `--runtime-version` property to `az functionapp create`

ARM

- `az deployment/group deployment validate`: Added `--handle-extended-json-format` parameter to support multiline and comments in json template when deployment.
- Bumped azure-mgmt-resource to 2019-07-01

Backup

- Added AzureFiles backup support

Compute

- `az vm create`: Added warning when specifying accelerated networking and an existing NIC together.

- `az vm create`: Added `--vmss` to specify an existing virtual machine scale set that the virtual machine should be assigned to.
- `az vm/vmss create`: Added a local copy of image alias file so that it can be accessed in a restricted network environment.
- `az vmss create`: Added `--orchestration-mode` to specify how virtual machines are managed by the scale set.
- `az vm/vmss update`: Added `--ultra-ssd-enabled` to allow updating ultra SSD setting.
- [BREAKING CHANGE] `az vm extension set`: Fixed bug where users could not set an extension on a VM with `--ids`.
- Added new commands `az vm image terms accept/cancel/show` to manage Azure Marketplace image terms.
- Updated VMAccessForLinux to version 1.5

CosmosDB

- [BREAKING CHANGE] `az sql container create`: Changed `--partition-key-path` to required parameter
- [BREAKING CHANGE] `az gremlin graph create`: Changed `--partition-key-path` to required parameter
- `az sql container create`: Added `--unique-key-policy` and `--conflict-resolution-policy`
- `az sql container create/update`: Updated the `--idx` default schema
- `gremlin graph create`: Added `--conflict-resolution-policy`
- `gremlin graph create/update`: Updated the `--idx` default schema
- Fixed typo in help message
- database: Added deprecation information
- collection: Added deprecation information

IoT

- Added new routing source type: DigitalTwinChangeEvents
- Fixed missing features in `az iot hub create`

Key Vault

- Fixed an unexpected error when certificate file does not exist
- Fixed `az keyvault recover/purge` not working

NetAppFiles

- Upgraded azure-mgmt-netapp to 0.6.0 to use API version 2019-07-01. This new API version includes:
 - Volume creation `--protocol-types` accepts now "NFSv4.1" not "NFSv4"
 - Volume export policy property now named 'nfsv41' not 'nfsv4'
 - Volume `--creation-token` renamed to `--file-path`
 - Snapshot creation date now named just 'created'

Network

- `az network private-dns link vnet create/update`: Support cross-tenant virtual network linking.
- [BREAKING CHANGE] `az network vnet subnet list`: Changed `--resource-group` and `--vnet-name` to be required now.
- `az network public-ip prefix create`: Added support to specify IP address version (IPv4, IPv6) when creation
- Bumped azure-mgmt-network to 7.0.0 and api-version to 2019-09-01
- `az network vrouter`: Added support for new service virtual router and virtual router peering
- `az network express-route gateway connection`: Added support for `--internet-security`

Profile

- Fixed `az account get-access-token --resource-type ms-graph` not working
- Removed warning from `az login`

RBAC

- Fixed `az ad app update --id {} --display-name {}` doesn't work

ServiceFabric

- `az sf cluster create`: Fixed an issue by modifying service fabric linux and windows template.json compute vmss from standard to managed disks

SQL

- Added `--compute-model`, `--auto-pause-delay`, and `--min-capacity` parameters to support CRUD operations for new SQL Database offering: Serverless compute model.

Storage

- `az storage account create/update`: Added `--enable-files-adds` parameter and Azure Active Directory Properties Argument group to support Azure Files Active Directory Domain Service Authentication
- Expanded `az storage account keys list/renew` to support listing or regenerating Kerberos keys of storage account.

October 15, 2019

Version 2.0.75

AKS

- Changed `--load-balancer-sku` default value to `standard` if supported by the kubernetes version
- Changed `--vm-set-type` default value to `virtualmachinescalesets` if supported by the kubernetes version

AMS

- [BREAKING CHANGE] Changed the name of `job start` to `job create`
- [BREAKING CHANGE] Changed the `--ask` parameter of `content-key-policy create` to use a 32-character hex string instead of UTF8

AppService

- Added commands `webapp config access-restriction show|set|add|remove`
- Added better error handling to `webapp up`
- Added support for `Isolated` SKU to `appservice plan update`

ARM

- Added `--handle-extended-json-format` parameter `deployment create` to support multiline and comments in json template

Compute

- Added `--enable-agent` parameter to `vm create`
- Changed `vm create` to use standard public IP SKU automatically when using zones
- Changed `vm create` to automatically create a valid computer name for a VM if none is provided
- Added `--computer-name-prefix` parameter to `vmss create` to support custom computer name prefix of virtual machines in the VMSS
- Add `--workspace` parameter to `vm create` to enable log analytics workspace automatically
- Updated galleries API version to 2019-07-01

Core

- Added syntax check for `--set` parameter in generic update command

IoT

- Fixed an issue where `iot hub show` would incorrectly error with "resource not found"

Monitor

- Added support for CRUD to `monitor log-analytics workspace`

Network

- Added support for cross-tenant virtual linking to `network private-dns link vnet [create|update]`
- [BREAKING CHANGE] Changed `network vnet subnet list` to require `--resource-group` and `--vnet-name` parameters

SQL

- Added commands to `sql mi ad-admin` that support setting an AAD administrator on managed instances

Storage

- Added `--preserve-s2s-access-tier` parameter to `storage copy` to preserve access tier during service to service copy
- Added `--enable-large-file-share` parameter to `storage account [create|update]` to support large file shares for storage account

September 24, 2019

Version 2.0.74

ACR

- Added a required `--type` parameter to `acr config retention update`
- [BREAKING CHANGE] Renamed parameter `--name -n` changed to `--registry -r` for `acr config` command group

AKS

- Added `--load-balancer-sku` parameter to `aks create` command, which allows for creating AKS cluster with SLB
- Added `--load-balancer-managed-outbound-ip-count`, `--load-balancer-outbound-ips` and `--load-balancer-outbound-ip-prefixes` parameters to `aks [create|update]` commands, which allow for updating load balancer profile of an AKS cluster with SLB
- Added `--vm-set-type` parameter to `aks create` command, which allows to specify vm types of an AKS Cluster (vmas or vmss)

ARM

- Added `--handle-extended-json-format` parameter to `group deployment create` command to support multiline and comments in json template

Compute

- Added `--terminate-notification-time` parameter to `vmss [create|update]` commands to support terminate scheduled event configurability
- Added `--enable-terminate-notification` parameter to `vmss update` command to support terminate scheduled event configurability
- Added `--priority`, `--eviction-policy`, `--max-billing` parameters to `[vm|vmss] create` commands

- Changed `disk create` to allow specifying the exact size of the disk upload
- Added support for incremental snapshots for managed disks to `snapshot create`

Cosmos DB

- Added `--type <key-type>` parameter to `cosmosdb keys list` command to show key, read only keys or connection strings
- Added `cosmosdb keys regenerate` command
- [DEPRECATED] Deprecated `cosmosdb list-connection-strings`, `cosmosdb regenerate-key` and `cosmosdb list-read-only-keys` commands

EventGrid

- Fixed the endpoint help text to refer to the right parameter

Key Vault

- Fixed issue where logging in with a tenant (`login -t`) could cause `keyvault create` to fail

Monitor

- Fixed issue where `:` character was not allowed in `--condition` argument to `monitor metrics alert create`

Policy

- Added support for Policy API version 2019-06-01
- Added `--enforcement-mode` parameter to `policy assignment create` command

Storage

- Added `--blob-type` parameter to `az storage copy` command

September 10, 2019

ACR

- Added command group `acr config retention` to configure retention policy

AKS

- Added support for ACR integration with the following commands:
 - Added `--attach-acr` parameter to `aks [create|update]` to attach an ACR to an AKS cluster
 - Added `--detach-acr` parameter to `aks update` to detach the ACR from an AKS cluster

ARM

- Updated to use API version 2019-05-10

Batch

- Added new JSON configuration settings to `--json-file` for `batch pool create`:
 - Added `MountConfigurations` for file system mounts (see [Request Body](#) for details)
 - Added optional property `publicIPs` on `NetworkConfiguration` for public IPs on pools (see [Request Body](#) for details)
- Added support for shared image galleries to `--image`
- [BREAKING CHANGE] Changed default value of `--start-task-wait-for-success` on `batch pool create` to be `true`
- [BREAKING CHANGE] Changed default value for `Scope` on `AutoUserSpecification` to always be Pool (was `Task` on Windows nodes, `Pool` on Linux nodes)
 - This argument can only be set from a JSON configuration with `--json-file`

HDInsight

- GA release
- [BREAKING CHANGE] Changed parameter `--workernode-count/-c` of `az hdinsight resize` to be required.

Key Vault

- Fixed issue where subnets couldn't be deleted from network rules
- Fixed issue where duplicated subnets and IP addresses could be added to network rules

Network

- Added `--interval` parameter to `network watcher flow-log` to set traffic analysis interval value
- Added `network application-gateway identity` to manage gateway identity
- Added support for setting Key Vault ID to `network application-gateway ssl-cert`
- Added `network express-route peering peer-connection [show|list]`

Policy

- Updated to use API version 2019-01-01

August 27, 2019

Version 2.0.72

ACR

- [BREAKING CHANGE] Removed support for the `classic` SKU

API Management

- [PREVIEW] Added `apim` command group

AppService

- Fixed issue with `webapp webjob continuous start` command when specifying a slot
- Changed `webapp up` to detect `env` folder and remove it from the file used for deployment

Keyvault

- Fixed a bug in `keyvault secret set` that ignored the `--expires` argument

Network

- Added support for IPv6 addresses to `--private-ip-address-version` arguments
- Added new commands `network private-endpoint [create|update|list-types]` for private endpoint management

- Added command group `network private-link-service`
- Added `--private-endpoint-network-policies` and `--private-link-service-network-policies` arguments to `network vnet subnet update`

RBAC

- Fixed issue with `ad app update --homepage` where homepage would not be updated

ServiceFabric

- Added support for mixed-case Key Vault names
- Fixed issue when using certificates in Key Vault
- Fixed issue with using PFX certificate files
- Fixed issue with `sf cluster certificate add` when Key Vault resource group wasn't specified
- Fixed issue with `sf cluster set` not working

SignalR

- Added new commands:
 - `signalr cors`: Manage SignalR CORS
 - `signalr restart`: Restart a SignalR service
 - `signalr update`: Update a SignalR service
- Added `--service-mode` argument to `signalr create`

Storage

- Added `storage account revoke-delegation-keys` command

August 13, 2019

Version 2.0.71

AppService

- Fixed issue where `webapp webjob continuous` commands were failing for slots

BotService

- [BREAKING CHANGE] Removed support for creating v3 SDK bots

CognitiveServices

- Added `cognitiveservices account network-rule` commands

Cosmos DB

- Removed warning when updating multiple write locations
- Added CRUD commands for CosmosDB SQL, MongoDB, Cassandra, Gremlin and Table resources and resource's throughput

HDInsight

This release contains a large number of breaking changes.

- [BREAKING CHANGE] Renamed parameters for `hdinsight create`:
 - Renamed `--storage-default-container` to `--storage-container`
 - Renamed `--storage-default-filesystem` to `--storage-filesystem`
- [BREAKING CHANGE] Changed the `--name` argument of `application create` to represent the application name instead of the cluster name
- Added `--cluster-name` argument to `application create` to replace old `--name` functionality
- [BREAKING CHANGE] Renamed parameters for `application create`:
 - Renamed `--application-type` to `--type`
 - Renamed `--marketplace-identifier` to `--marketplace-id`
 - Renamed `--https-endpoint-access-mode` to `--access-mode`
 - Renamed `--https-endpoint-destination-port` to `--destination-port`
- [BREAKING CHANGE] Removed parameters for `application create`:
 - `--https-endpoint-location`
 - `--https-endpoint-public-port`
 - `--ssh-endpoint-destination-port`
 - `--ssh-endpoint-location`
 - `--ssh-endpoint-public-port`
- [BREAKING CHNAGE] Renamed `--target-instance-count` to `--workernode-count` for `hdinsight resize`

- [BREAKING CHANGE] Changed all commands in the `hdinsight script-action` group to use the `--name` parameter as the name of the script action.
- Added `--cluster-name` argument to all `hdinsight script-action` commands to replace old `--name` functionality
- [BREAKING CHANGE] Renamed `--script-execution-id` to `--execution-id` for all `hdinsight script-action` commands
- [BREAKING CHANGE] Renamed `hdinsight script-action show` to `hdinsight script-action show-execution-details`
- [BREAKING CHANGE] Changed parameters to `hdinsight script-action execute --roles` to be space-separated instead of comma-separated
- [BREAKING CHANGE] Removed the `--persisted` parameter of `hdinsight script-action list`
- Changed the `hdinsight create --cluster-configurations` parameter to accept a path to a local JSON file or a JSON string
- Added command `hdinsight script-action list-execution-history`
- Changed `hdinsight monitor enable --workspace` to accept a Log Analytics workspace ID or workspace name
- Added the `hdinsight monitor enable --primary-key` argument, which is needed if a workspace ID is provided as the parameter
- Added more examples and updated descriptions for help messages

Interactive

- Fixed a loading error

Kubernetes

- Changed to use `https` if dashboard container port is using `https`

Network

- Added `--yes` argument `network dns record-set cname delete`

Profile

- Added `--resource-type` argument to `account get-access-token` to get resource access tokens

ServiceFabric

- Added all supported os version for sf cluster create
- Fixed primary certificate validation bug

Storage

- Added command `storage copy`

July 30, 2019

Version 2.0.70

ACR

- Fixed issue #9952 (a regression in the `acr pack build` command)
- Removed the default builder image name in `acr pack build`

Appservice

- Changed `webapp config ssl` to show a message if a resource is not found
- Fixed issue where `functionapp create` does not accept `Standard_RAGRS` storage account type
- Fixed an issue where `webapp up` would fail if run using older versions of python

Network

- Removed invalid parameter `--ids` from `network nic ip-config add` (fixes #9861)
- Fixes #9604. Added `--root-certs` parameter to `network application-gateway http-settings [create|update]` to support user associate trusted root certificates.
- Fixed argument `--subscription` for `network dns record-set ns create` (#9965)

RBAC

- Added `user update` command
- [DEPRECATED] Deprecated `--upn-or-object-id` from user-related commands
 - Use replacement argument `--id`
- Added `--id` argument to user-related commands

SQL

- Added management commands for managed instance keys and TDE protector

Storage

- Added `storage remove` command
- Fixed an issue with `storage blob update`

VM

- Changed `list-skus` to use newer api-version to output zone details
- Changed default of `--single-placement-group` to `false` for `vmss create`
- Added ability to select ZRS storage SKUs for `[snapshot|disk] create`
- Added new command group `vm host` to support dedicated hosts
- Added parameters `--host` and `--host-group` on `vm create` to set VM dedicated host

July 16, 2019

Version 2.0.69

Appservice

- Changed `webapp identity` commands to return a proper error message if ResourceGroupName or App name are invalid
- Fixed `webapp list` to return the correct value for numberOfSites if no ResourceGroup was provided
- Fixed side-effects of `appservice plan create` and `webapp create`

Core

- Fixed issue where `--subscription` would appear despite being not applicable

Batch

- [BREAKING CHANGE] Replaced `batch pool node-agent-skus list` with `batch pool supported-images list`

- Added support for security rules blocking network access to a pool based on the source port of the traffic when using the `--json-file` option of `batch pool create network`
- Added support for executing the task in the container working directory or in the Batch task working directory when using the `--json-file` option of `batch task create`
- Fixed error in `--application-package-references` option of `batch pool create` where it would only work with defaults

Eventhubs

- Added validation for parameter `--rights` of `authorizationrule` commands

RDBMS

- Added optional parameter to specify replica SKU for create replica command
- Fixed the issue with CI test failure with creating MySQL replica

Relay

- Fixed issue with hybrid connection when client authroization disabled [#8775](#)
- Added parameter `--requires-transport-security` to `relay wcfrelay create`

Servicebus

- Added validation for parameter `--rights` of `authorizationrule` commands

Storage

- Enable Files AADDS for storage account update
- Fixed issue `storage blob service-properties update --set`

July 2, 2019

Version 2.0.68

Core

- Command modules are now consolidated into a single Python distributable. This deprecates direct use of many `azure-cli-` packages on PyPI. This should reduce install size and only affect users who have directly installed via `pip`.

ACR

- Added support for Timer Triggers to Task

Appservice

- Changed `functionapp create` to enable application insights by default
- [BREAKING CHANGE] Removed deprecated `functionapp devops-build` command.
 - Use the new command `az functionapp devops-pipeline` instead
- Added Linux Consumption function app plan support to `functionapp deployment config-zip`

Cosmos DB

- Added support for disabling TTL

DLS

- Updated ADLS version (0.0.45)

Feedback reference

- When reporting a failed extension command, `az feedback` now attempts to open the browser to the project/repo url of the extension from the index

HDInsight

- [BREAKING CHANGE] Changed `oms` command group name to `monitor`
- [BREAKING CHANGE] Made `--http-password/-p` a required parameter
- Added completers for `--cluster-admin-account` and `cluster-users-group-dns` parameters completer
- Changed `cluster-users-group-dns` parameter to be required when `-esp` is present
- Added a timeout for all existing argument auto-completers
- Added a timeout for transforming resource name to resource id

- Changed Auto-completers to select resources from any resource group. It can be a different resource group than the one specified with `-g`
- Added support for `--sub-domain-suffix` and `--disable_gateway_auth` parameters in the `hdinsight application create` command

Managed Services

- Introducing managed service command module in preview

Profile

- Suppress `--subscription` argument for logout command

RBAC

- [BREAKING CHANGE] Removed `--password` argument for `create-for-rbac`
- Added `--assignee-principal-type` parameter to `create` command to avoid intermittent failures caused by AAD graph server replication latency
- Fixed a crash in `ad signed-in-user` when listing owned objects
- Fixed issue where `ad sp` would not find the right application from a service principal

RDBMS

- Added support for replication for MariaDB

SQL

- Documented allowed values for `sql db create --sample-name`

Storage

- Added user delegation SAS token support with `--as-user` to `storage blob generate-sas`
- Added user delegation SAS token support with `--as-user` to `storage container generate-sas`

VM

- Fixed bug where `vmss create` returns an error message when run with `--no-wait`
- Removed client-side validation for `vmss create --single-placement-group`. Does not fail if `--single-placement-group` is set to `true` and `--instance-count` is greater than 100 or availability zones are specified, but leaves this validation to the compute service
- Fixed bug where `[vm|vmss] extension image list` fails when used with `--latest`

June 18, 2019

Version 2.0.67

Core

This release introduces a new [Preview] tag to more clearly communicate to customers when a command group, command or argument is in preview status. This was previously called out in help text or communicated implicitly by the command module version number. The CLI will be removing version numbers for individual packages in the future. If a command is in preview, all of its arguments are as well. If a command group is labeled as being in preview, then all commands and arguments are considered to be in preview as well.

As a result of this change, several command groups may seem to "suddenly" appear to be in a preview status with this release. What actually happened is that most packages were in a preview status, but are being deemed GA with this release

ACR

- Added 'acr check-health' command
- Improved error handling for AAD tokens and for retrieving external commands

ACS

- Deprecated ACS commands are now hidden from help view

AMS

- [BREAKING CHANGE] Changed to return ISO 8601 time strings for archive-window-length and key-frame-interval-duration

AppService

- Added location based routing for `webapp deleted list` and `webapp deleted restore`
- Fixed issue where webapp up logged target URL ("You can launch the app at...") was not clickable in Azure Cloud Shell
- Fixed an issue where creating apps with the some SKUs was failing with an AlwaysOn error
- Added pre-validation to `[appservice|webapp] create`
- Fixed `[webapp|functionapp] traffic-routing` to use the correct actionHostName
- Added slot support to `functionapp` commands

Batch

- Fixed AAD auth regression caused by over-aggressive error reporting for Shared Key Auth

BatchAI

- BatchAI commands are now deprecated and hidden

BotService

- Added "discontinued support"/"maintenance mode" warning messages for commands that support the v3 SDK

CosmosDB

- [DEPRECATED] Deprecated the `cosmosdb list-keys` command
- Added the `cosmosdb keys list` command - replaces `cosmosdb list-keys`
- `cosmosdb create/update`: Added new format for --location to allow setting "isZoneRedundant" property. Deprecated old format

EventGrid

- Added `eventgrid domain` commands for domain CRUD operations
- Added `eventgrid domain topic` commands for domain topics CRUD operations
- Added `--odata-query` argument to `eventgrid [topic|event-subscription] list` for filtering results using OData syntax
- `event-subscription create/update`: Added servicebusqueue as new values for the `--endpoint-type` parameter

- [BREAKING CHANGE] Removed support for `--included-event-types All` with
`eventgrid event-subscription [create|update]`

HDInsight

- Added support for `--ssh-public-key` parameter in `hdinsight create` command

IoT

- Added support to regenerate authorization policy keys
- Added SDK and support for DigitalTwin Repository Provisioning Service

Network

- Added Zone support for Nat Gateway
- Added command `network list-service-tags`
- Fixed issue with `dns zone import` where users could not import wildcard A records
- Fixed issue with `watcher flow-log configure` where flow logging could not be enabled in certain regions

Resource

- Added `az rest` command for making REST calls
- Fixed error when using `policy assignment list` with a resource group or subscription level `--scope`

ServiceBus

- Fixed issue with `servicebus topic create --max-size #9319 ↗`

SQL

- Changed `--location` to be optional for `sql [server|mi] create` - uses resource group location if not specified
- Fixed "'NoneType' object is not iterable" error for `sql db list-editions --available`

SQLVm

- [BREAKING CHNAGE] Changed `sql vm create` to require `--license-type` parameter
- Changed to allow setting SQL image SKU when creating or updating a sql vm

Storage

- Fixed issue with missing account key for `storage container generate-sas`
- Fixed issue with `storage blob sync` on Linux

VM

- [PREVIEW] Added `vm image template` commands to build VM images

June 4, 2019

Version 2.0.66

Core

- Fixed bug where commands fail if `--output yaml` is used with `--query`

ACR

- Added 'acr pack' command group for creating quick build Tasks using Buildpacks.

ACS

- Allow enabling/disabling AKS kube-dashboard addon
- Print a friendly message when the subscription is not approved to use Azure Red Hat OpenShift

Batch

- Improved error handling when not logged in to an account [[#9165](#)][[#8978](#)]

IoT

- Added support for manual failover

Network

- Added `network application-gateway waf-policy` commands to support custom WAF rules.
- Added `--waf-policy` and `--max-capacity` arguments to `network application-gateway [create|update]`

Resource

- Improved error message from `deployment create` when there is no TTY available

Role

- Updated help text.

Compute

- Added support to `vm create` for VMs from a managed image with data-disk luns that do not start from 0 or that skip numbers

May 21, 2019

Version 2.0.65

Core

- Added better feedback for authentication errors
- Fixed issue where the CLI would load extensions that were not compatible with its core version
- Fixed issue with launching when `clouds.config` is corrupted

ACR

- Added support for Managed Identities to Tasks

ACS

- Fixed `openshift create` command when used with customer AAD client

AppService

- [DEPRECATED] Deprecated `functionapp devops-build` command - will be removed in next release
- Changed `functionapp devops-pipeline` to fetch build log from Azure DevOps in verbose mode
- [BREAKING CHANGE] Removed `--use_local_settings` flag from `functionapp devops-pipeline` command - was a no-op
- Changed `webapp up` to return JSON output if `--logs` is not used
- Added support for writing default resources to local config for `webapp up`
- Added support to `webapp up` for redeploying an app without using the `--location` argument
- Fixed an issue where for Linux Free SKU ASP creation use Free as SKU value was not working

BotService

- Changed to allow all casing for `--lang` parameters for commands
- Updated description for command module

Consumption

- Added missing required parameter when running `consumption usage list --billing-period-name`

IoT

- Added support to list all keys

Network

- [BREAKING CHANGE]: Removed `network interface-endpoints` command group - use `network private-endpoints`
- Added `--nat-gateway` argument to `network vnet subnet [create|update]` for attaching to a NAT gateway
- Fixed issue with `dns zone import` where record names could not match a record type

RDBMS

- Added postgres and mysql support for geo replication

RBAC

- Added support for management group scope to `role assignment`

Storage

- `storage blob sync`: add sync command for storage blob

Compute

- Added `--computer-name` to `vm create` for setting a VM's computer name
- Renamed `--ssh-key-value` renamed to `--ssh-key-values` for `[vm|vmss] create` - can now accept multiple ssh public key values or paths
 - **Note:** This is **not** a breaking change - `--ssh-key-value` will be parsed correctly as it matches only `--ssh-key-values`
- Changed the `--type` argument of `ppg create` to be optional

May 6, 2019

Version 2.0.64

ACS

- [BREAKING CHANGE] Removed `--fqdn` flag on `openshift` commands
- Changed to use Azure Red Hat Openshift GA API Version
- Added `customer-admin-group-id` flag to `openshift create`
- [GA] Removed `(PREVIEW)` from `aks create` option `--network-policy`

Appservice

- [DEPRECATED] Deprecated `functionapp devops-build` command
 - Renamed to `functionapp devops-pipeline`
- Fixed getting the correct username for cloudshell which was causing `webapp up` to fail
- Updated `appservice plan --sku` documentation updated to reflect the supported appserviceplans
- Added optional arguments for resource group and plan to `webapp up`

- Added support to `webapp ssh` to respect `AZURE_CLI_DISABLE_CONNECTION_VERIFICATION` environment variable
- Added `appserviceplan create` support for Linux Free SKU
- Changed `webapp up` to have a 30s sleep after setting `SCM_DO_BUILD_DURING_DEPLOYMENT=true` appsetting to handle kudu cold start
- Added support for `powershell` runtime to `functionapp create` on Windows
- Added `create-remote-connection` command

Batch

- Fixed bug in validator for `--application-package-references` options

Botservice

- [BREAKING CHANGE] Changed `bot create -v v4 -k webapp` to create an empty Web App Bot by default (i.e. no bot is deployed to the App Service)
- Added `--echo` flag to `bot create` to use the old behavior with `-v v4`
- [BREAKING CHANGE] Changed the default value of `--version` to `v4`
 - **NOTE:** `bot prepare-publish` still uses its old default
- [BREAKING CHANGE] Changed `--lang` to no longer default to `Csharp`. If the command requires `--lang` and it is not provided, the command will now error out
- [BREAKING CHANGE] Changed the `--appid` and `--password` args for `bot create` to be required and can now be created via `ad app create`
- Added `--appid` and `--password` validation
- [BREAKING CHANGE] Changed `bot create -v v4` to not create or use a Storage Account or Application Insights
- [BREAKING CHANGE] Changed `bot create -v v3` to require a region where Application Insights is available
- [BREAKING CHANGE] Changed `bot update` to now affect only specific properties of a bot
- [BREAKING CHANGE] Changed `--lang` flags to accept `Javascript` instead of `Node`
- [BREAKING CHANGE] Removed `Node` as an allowed `--lang` value
- [BREAKING CHANGE] Changed `bot create -v v4 -k webapp` to no longer set `SCM_DO_BUILD_DURING_DEPLOYMENT` to true. All deployments through Kudu will act according to their default behavior
- Changed `bot download` for bots without `.bot` files to create the language-specific configuration file with values from the Application Settings for the bot
- Added `Typescript` support to `bot prepare-deploy`

- Added warning message to `bot prepare-deploy` for `Javascript` and `Typescript` bots for when `--code-dir` does not contain `package.json`
- Changed `bot prepare-deploy` to return `true` if successful
- Added verbose logging to `bot prepare-deploy`
- Added more available Application Insights regions to `az bot create -v v3`

Configure

- Added support for folder based argument default value configurations

Eventhubs

- Added `namespace network-rule` commands
- Added `--default-action` argument for network rules to `namespace [create|update]`

Network

- [BREAKING CHANGE] Replaced `--cache` argument with `--defer` for `vnet [create|update]`

Policy Insights

- Added support for `--expand PolicyEvaluationDetails` to query policy evaluation details on the resource

Role

- [DEPRECATED] Changed `create-for-rbac` hide '--password' argument - support will be removed in May 2019

Service Bus

- Added `namespace network-rule` commands
- Added `--default-action` argument for network rules to `namespace [create|update]`
- Fixed `topic [create|update]` to allow `--max-size` support for 10, 20, 40 and 80GB values with premium SKU

SQL

- Added `sql virtual-cluster [list|show|delete]` commands

VM

- Added `--protect-from-scale-in` and `--protect-from-scale-set-actions` to `vmss update` to enable updates to the protection policy of VMSS VM instances
- Added `--instance-id` to `vmss update` to enable generic update of VMSS VM instances
- Added `--instance-id` to `vmss wait`
- Added new `ppg` command group for managing Proximity Placement Groups
- Added `--ppg` to `[vm|vmss] create` and `vm availability-set create` for managing PPGs
- Added `--hyper-v-generation` parameter to `image create`

April 23, 2019

Version 2.0.63

ACS

- Changed `aks get-credentials` to prompt to overwrite duplicated values
- Removed `(PREVIEW)` from Dev Spaces commands "aks use-dev-spaces" and "aks remove-dev-spaces"

AMS

- Fixed bug with asset and account filters update

AppService

- Added support for ASE and timeout to `webapp ssh`
- Added support for establishing CI CD to an Azure DevOps pipeline from a Github repository to Function apps
- Added `--github-pat` argument to `functionapp devops-build create` to accept Github personal access token
- Added `--github-repository` argument to `functionapp devops-build create` to accept Github repository that contains a functionapp source code

- Fixed issue where `az webapp up --logs` was failing with a error and updating default .NETCORE version to 2.1
- Removed unnecessary functionapp settings when creating a function app with consumption plan
- Changed `webapp up` so the default asp string now appends number at the end to create a new ASP based on SKU options
- Added `-b` as an option to `webapp up` to launch the app in the browser
- Changed `webapp deployment source config zip` to handle `AZURE_CLI_DISABLE_CONNECTION_VERIFICATION` environment variable

Deployment Manager

- [PREVIEW] Create and manage artifacts that support rollouts

Lab

- Fixed bug which would cause an early exit

Network

- Added auto name server delegation to `dns zone create` in parent during child zone creation

Resource

- [DEPRECATED] Deprecated `--link-id`, `--target-id` and `--filter-string` arguments of `resource link`
 - Use the arguments `--link`, `--target`, and `--filter` instead
- Fixed issue where `resource link [create|update]` commands would not work
- Fixed an issue where deleting using a resource ID could crash on error

SQL

- Added support for custom time zone on managed instances
- Changed to allow elastic pool name to be used with `sql db update`
- Added `--no-wait` support to `sql server [create|update]`
- Added command `sql server wait`

Storage

- Fixed issue with double-encoded SAS tokens in `storage blob generate-sas`

VM

- Added `--skip-shutdown` flag to `vm|vmss stop` to power-off VMs without shutdown
- Added `--storage-account-type` argument to `sig image-version create` to set the publishing profile's account type
- Added `--target-regions` argument to `sig image-version create` to allow setting region-specific storage account types

April 9, 2019

Core

- Fixed issue where some extensions showed a version of `Unknown` and could not be updated

ACR

- Added support running an image contextlessly

AMS

- [DEPRECATED]: Deprecated the `--bitrate` parameter of `account-filter` and `asset-filter`
- [BREAKING CHANGE]: Renamed the `--bitrate` parameter to `--first-quality`
- Added new encryption parameters support in `ams streaming-policy create`
- Added new parameter `--filters` to `ams streaming-locator create`

AppService

- Added `--logs` support to `webapp up`
- Fixed `functionapp devops-build create` command `azure-pipelines.yml` generation issues
- Improved `unctionapp devops-build create` error handling and indicators
- [BREAKING CHANGE] Removed the `--local-git` flag for `devops-build` command, local git detection and handling are compulsory for creating Azure DevOps pipelines

- Added support for Linux functions plan creation
- Added ability to switch a plan underneath a function app using `functionapp update --plan`
- Added support for Azure Functions premium plan scale out settings

CDN

- Added support for `Microsoft_Standard` and `Standard_ChinaCdn`

Feedback reference

- Changed `feedback` to show metadata on recently run commands
- Changed `feedback` to prompt user to assist in issue creation process by opening a browser and using an issue template
- Changed `feedback` to print out issue body when run with '--verbose'

Monitor

- Fixed issue where "count" was not a permitted value with `metrics alert [create|update]`

Network

- Fixed table format not displaying with `vnet-gateway list-bgp-peer-status`
- Added `list-request-headers` and `list-response-headers` commands to `application-gateway rewrite-rule`
- Added `list-server-variables` command to `application-gateway rewrite-rule condition`
- Fixed an issue where updating link state on an express-route port would throw an unknown attribute exception `express-route port update`

PrivateDNS

- Added `network private-dns` for Private DNS zones

Resource

- Fixed issue with `deployment create` and `group deployment create` where a parameters file with an empty set of parameters would not work

Role

- Fixed `create-for-rbac` to handle `--years` correctly
- [BREAKING CHANGE] Changed `role assignment delete` to prompt when deleting all assignments under the subscription unconditionally

SQL

- Updated `sql mi [create|update]` with the properties `proxyOverride` and `publicDataEndpointEnabled`

Storage

- [BREAKING CHANGE] Removed result of `storage blob delete`
- Added `--full-uri` to `storage blob generate-sas` to create the full uri for the blob with sas
- Added `--file-snapshot` to `storage file copy start` to copy file from snapshot
- Changed `storage blob copy cancel` to only show the error instead of exception for `NoPendingCopyOperation`

March 26, 2019

Core

- Fixed issues with dev extension incompatibility
- Error handling now points customers to issues page

Cloud

- Fixed a 'subscription not found' error in `cloud set`

ACR

- Fixed redundant sources in image import
- Added `--auth-mode` to `acr build`, `acr run`, `acr task create`, and `acr task update` commands
- Added 'acr task credential' command group for managing credentials for a Task
- Added '--no-wait' to `acr build` command

AppService

- Fixed bug where `webapp up` was not handling running from empty directory or unknown code scenario correctly
- Fixed bug where slots didn't work for `[webapp|functionapp] config ssl bind`

BOT Service

- Added `bot prepare-deploy` to prepare for deploying bots via `webapp`
- Changed `bot create --kind registration` to show password if the password is not provided
- [BREAKING CHANGE] Changed `--endpoint` in `bot create --kind registration` to default to an empty string instead of being required
- Added `SCM_DO_BUILD_DURING_DEPLOYMENT` to ARM template's Application Settings for v4 Web App Bots

CDN

- Added support for `--no-wait` to `cdn endpoint`
[`create|update|start|stop|delete|load|purge`]
- [BREAKING CHANGE]: Changed `cdn endpoint create` default query string caching behaviour. No longer defaults to "IgnoreQueryString". It is now set by the service

Cosmosdb

- Added support for `--enable-multiple-write-locations` on account update
- Added `network-rule` subgroup with commands `add`, `remove`, and `list` for managing VNET rules of a Cosmos DB account

Interactive

- Fixed incompatibility with Interactive extension installed through azdev

Monitor

- Changed to allow dimension value `*` for `monitor metrics alert [create|update]`

Network

- Added `rewrite-rule` command group to `application-gateway`

Profile

- Added tenant level account support for managed service identity to `login`

Postgres

- Added postgresql `replica` commands and `restart server` command
- Changed to get default location from resource group when not provided for creating servers and add validation for retention days

Resource

- Improved table output for `deployment [create|list|show]`
- Fixed issue with `deployment [create|validate]` where type `secureObject` was not recognized

Graph

- Added support for `--end-date` to `ad [app|sp] credential reset`
- Added support to add permissions with `ad app permission add`
- Fixed a bug with `ad app permission list` when there were no permissions
- Changed `ad sp delete` to skip role assignment delete if the current account has no subscription
- Changed `ad app create` to have `--identifier-uris` default to empty list if not provided

storage

- Added `--snapshot` to `storage file download-batch` to download from a share snapshot
- Changed `storage blob [download-batch|upload-batch]` progress bar to be less verbose and indicate current blob
- Fixed issue with `storage account update` when updating encryption parameters
- Fixed issue where `storage blob show` would fail when using oauth (`--auth-mode=login`)

VM

- Added `image update` command

March 12, 2019

Version 2.0.60

Core

- Fixed an incorrect error in `cloud set` about subscription not found

ACR

- Fixed redundant sources in image import

ACS

- Changed to ignore the `--listen-address` parameter for `aks browse` if it is not supported by kubectl

AppService

- Added `[webapp|functionapp] deployment list-publishing-credentials` to get the Kudu publishing url and its credentials
- Removed erroneous print statement for `webapp auth update`
- Fixed `functionapp` to set the correct image for runtime in Linux App Service plans
- Removed preview tag for `webapp up` and added improvements to the command

Botservice

- Added `SCM_DO_BUILD_DURING_DEPLOYMENT` to ARM template's Application Settings for v4 Web App Bots
- Added `Microsoft-BotFramework-AppId` and `Microsoft-BotFramework-AppPassword` to ARM template's Application Settings for v4 Web App Bots
- Removed single quotes from `bot publish` command output at end of `bot create`
- Changed `bot publish` to be asynchronous

Container

- Added `--no-wait` argument to `container [start|restart]`

EventHub

- Added `--skip-empty-archives` flag to `eventhub create|update` to support empty archives in capture

Find

- Major functionality update

HDInsight

- Added the `--storage-account-managed-identity` parameter to `hdinsight create` to support ADLS Gen2 MSI

Network

- Fixed issue with `vpn-connection update` where updating a VPN connection between gateways in different subscriptions would fail

Rdbms

- Minor fixes to get default location from resource group when not provided for creating servers and add validation for retention days

Role

- Fixed `role definition update` to use ID to resolve definition correctly
- Changed `ad app credential reset` to remove the assumption that app's service principal always exists

Service Fabric

- Fixed issue with `sf cluster list` was not iterable

February 26, 2019

Version 2.0.59

Core

- Fixed issue where in some instances using `--subscription NAME` would throw an exception

ACR

- Added `--target` parameter for `acr build`, `acr task create` and `acr task update` commands
- Improved error handling for runtime commands when not logged into Azure

ACS

- Added `--listen-address` option to `aks port-forward`

AppService

- Added `functionapp devops-build` command

Batch

- [BREAKING CHANGE] Removed the `batch pool upgrade os` command
- [BREAKING CHANGE] Removed the `Pacakges` property from `Application` responses
- Added the `batch application package list` command to list packages of an application
- [BREAKING CHANGE] Changed `--application-id` to `--application-name` in all `batch application` commands,
- Added the `--json-file` argument to commands for requesting the raw API response
- Updated validation to automatically include `https://` in all endpoints if missing

CosmosDB

- Added `network-rule` subgroup with commands `add`, `remove`, and `list` for managing VNET rules of a Cosmos DB account

Kusto

- [BREAKING CHANGE] Changed `hot_cache_period` and `soft_delete_period` types for database to ISO8601 duration format

Network

- Added `--express-route-gateway-bypass` argument to `vpn-connection [create|update]`
- Added command groups from `express-route` extensions
- Added `express-route gateway` and `express-route port` command groups
- Added argument `--legacy-mode` to `express-route peering [create|update]`
- Added arguments `--allow-classic-operations` and `--express-route-port` to `express-route [create|update]`
- Added `--gateway-default-site` argument to `vnet-gateway [create|update]`
- Added `ipsec-policy` commands to `vnet-gateway`

Resource

- Fixed issue with `deployment create` where type field was case-sensitive
- Added support for URI-based parameters file to `policy assignment create`
- Added support for URI-based parameters and definitions to `policy set-definition update`
- Fixed handling of parameters and rules for `policy definition update`
- Fixed issue with `resource show/update/delete/tag/invoke-action` where cross-subscription IDs did not properly honor the subscription ID

Role

- Added support for app roles to `ad app [create|update]`

VM

- Fixed issue with `vm create` where `--accelerated-networking`` was not enabled by default for Ubuntu 18.0

February 12, 2019

Version 2.0.58

Core

- `az --version` now displays a notification if you have packages that can be updated

- Fixed regression where `--ids` could no longer be used with JSON output

ACR

- [BREAKING CHANGE] Removed `acr build-task` command group
- [BREAKING CHANGE] Removed `--tag` and `--manifest` options from `acr repository delete`

ACS

- Added support for case-insensitive names to `aks [enable-addons|disable-addons]`
- Added support for Azure Active Directory updating operation using `aks update-credentials --reset-aad`
- Added clarification that `--output` is ignored for `aks get-credentials`

AMS

- Added `ams streaming-endpoint [start | stop | create | update] wait` commands
- Added `ams live-event [create | start | stop | reset] wait` commands

Appservice

- Added ability to create and configure functions using ACR containers
- Added support for updating webapp configurations through json
- Improved help for `appservice-plan-update`
- Added support for app insights on functionapp create
- Fixed issues with webapp SSH

Botservice

- Improved UX for `bot publish`
- Added warning for timeouts when running `npm install` during `az bot publish`
- Removed invalid char `.` from `--name` in `az bot create`
- Changed to stop randomizing resource names when creating Azure Storage, App Service Plan, Function/Web App and Application Insights
- [DEPRECATED] Deprecated `--proj-name` argument in favor of `--proj-file-path`
- Changed `az bot publish` to remove fetched IIS Node.js deployment files if they did not already exist

- Added `--keep-node-modules` argument to `az bot publish` to not delete `node_modules` folder on App Service
- Added `"publishCommand"` key-value pair to output from `az bot create` when creating an Azure Function or Web App bot
 - The value of `"publishCommand"` is an `az bot publish` command prepopulated with the required parameters to publish the newly created bot
- Updated `"WEBSITE_NODE_DEFAULT_VERSION"` in ARM template for v4 SDK bots to use 10.14.1 instead of 8.9.4

Key Vault

- Fixed issue with `keyvault secret backup` where some users received an `unexpected_keyword` error when using `--id`

Monitor

- Changed `monitor metrics alert [create|update]` to allow dimension value `*`

Network

- Changed `dns zone export` to ensure exported CNAMEs are FQDNs
- Added `--gateway-name` parameter to `nic ip-config address-pool [add|remove]` to support application gateway backend address pools
- Added `--traffic-analytics` and `--workspace` arguments to `network watcher flow-log configure` to support traffic analytics through a Log Analytics workspace
- Added `--idle-timeout` and `--floating-ip` to `lb inbound-nat-pool [create|update]`

Policy Insights

- Added `policy remediation` commands to support resource policy remediation features

RDBMS

- Improved help message and command parameters

Redis

- Added commands for managing firewall-rules (create, update, delete, show, list)
- Added commands for managing server-link (create, delete, show, list)
- Added commands for managing patch-schedule (create, update, delete, show)
- Added support for Availability Zones and Minimum TLS Version to `redis create`
- [BREAKING CHANGE] Removed `redis update-settings` and `redis list-all` commands
- [BREAKING CHANGE] Parameter for `redis create`: 'tenant settings' is not accepted in key[=value] format
- [DEPRECATED] Added warning message for deprecating `redis import-method` command

Role

- [BREAKING CHANGE] Moved `az identity` command here from `vm` commands

SQL VM

- [DEPRECATED] Deprecated `--bootstrap-acc-pwd` argument due to typo

VM

- Changed `vm list-skus` to allow use of `--all` in place of `--all true`
- Added `vmss run-command [invoke | list | show]`
- Fixed bug where `vmss encryption enable` would fail if run previously
- [BREAKING CHANGE] Moved `az identity` command to `role` commands

January 31, 2019

Version 2.0.57

Core

- Hot Fix for [issue 8399](#).

January 28, 2019

Version 2.0.56

ACR

- Added support for VNet/IP rules

ACS

- Added Virtual Nodes Preview
- Added Managed OpenShift commands
- Added support for service principal updates operation with `aks update-credentials -reset-service-principal`

AMS

- [BREAKING CHANGE] Renamed `ams asset get-streaming-locators` to `ams asset list-streaming-locators`
- [BREAKING CHANGE] Renamed `ams streaming-locator get-content-keys` to `ams streaming-locator list-content-keys`

Appservice

- Added support for app insights on `functionapp create`
- Added support for app service plan creation (including Elastic Premium) to Function Apps
- Fixed app setting issues with Elastic Premium plans

Container

- Added `container start` command
- Changed to allow using decimal values for CPU during container creation

EventGrid

- Added `--deadletter-endpoint` parameter to `event-subscription [create|update]`
- Added storagequeue and hybridconnection as new values for 'event-subscription [create|update] --endpoint-type'
- Added `--max-delivery-attempts` and `--event-ttl` parameters to `event-subscription create` to specify the retry policy for events
- Added a warning message to `event-subscription [create|update]` when webhook as destination is used for an event subscription

- Added source-resource-id parameter for all event subscription related commands and mark all other source resource related parameters as deprecated

HDInsight

- [BREAKING CHANGE] Removed the `--virtual-network` and `--subnet-name` parameters from `hdinsight [application] create`
- [BREAKING CHANGE] Changed `hdinsight create --storage-account` to accept name or id of storage account instead of blob endpoints
- Added `--vnet-name` and `--subnet-name` parameters to `hdinsight create`
- Added support for Enterprise Security Package and disk encryption to `hdinsight create`
- Added `hdinsight rotate-disk-encryption-key` command
- Added `hdinsight update` command

IoT

- Added encoding format to routing-endpoint command

Kusto

- Preview release

Monitor

- Changed ID comparison to be case insensitive

Profile

- Enable tenant level account for managed service identity for `login`

Network

- Fixed issue with `express-route update`: where `--bandwidth` argument was ignored
- Fixed issue with `ddos-protection update` where set comprehension caused stack trace

Resource

- Added support for URI parameters file to `group deployment create`
- Added support for managed identity to `policy assignment [create|list|show]`

SQL Virtual Machine

- Preview release

Storage

- Changed fix to update only properties that are changed on the same object
- Fixed #8021, binary data is encoded in base 64 when returned

VM

- Changed `vm encryption enable` to validate disk encryption keyvault and that key encryption keyvault exists
- Added `--force` flag to `vm encryption enable`

January 15, 2019

Version 2.0.55

ACR

- Changed to allow force push a helm chart that doesn't exist
- changed to allow runtime operations without ARM requests
- [DEPRECATED] Deprecated `--resource-group` parameter in the commands:
 - `acr login`
 - `acr repository`
 - `acr helm`

ACS

- Added support for new ACI regions

Appservice

- Fixed issue with uploading certificates for apps that are hosted on an ASE, where the ASE RG & App RG are different

- Changed `webapp up` to use SKU P1V1 as default for Linux
- Fixed `[webapp|functionapp] deployment source config-zip` to show the right error message when a deployment fails
- Added `webapp ssh` command

Botservice

- Added deployment status updates to `bot create`

Configure

- Added `none` as a configurable output format

CosmosDB

- Added support for creating database with shared throughput

HDInsight

- Added commands for managing applications
- Added commands for managing script actions
- Added commands for managing Operations Management Suite (OMS)
- Added support to list regional usage to `hdinsight list-usage`
- [BREAKING CHANGE] Removed default cluster type from `hdinsight create`

Network

- Added `--custom-headers` and `--status-code-ranges` arguments to `traffic-manager profile [create|update]`
- Added new routing types: Subnet and Multivalue
- Added `--custom-headers` and `--subnets` arguments to `traffic-manager endpoint [create|update]`
- Fixed issue where supplying `--vnets ""` to `ddos-protection update` caused an error

Role

- [DEPRECATED] Deprecated `--password` argument for `create-for-rbac`. Use secure passwords generated by the CLI instead

Security

- Initial Release

Storage

- [BREAKING CHANGE] Changed `storage [blob|file|container|share] list` default number of results to be 5,000. Use `--num-results *` for original behavior of returning all results
- Added `--marker` parameter to `storage [blob|file|container|share] list`
- Added log marker for next page to STDERR for `storage [blob|file|container|share] list`
- Added `storage blob service-properties update` command with support for static websites

VM

- Changed `vm [disk|unmanaged-disk]` and `vmss disk` to have more consistent parameters
- Added support for cross tenant image referencing to `[vm|vmss] create`
- Fixed bug with default configuration in `vm diagnostics get-default-config -- windows-os`
- Added argument `--provision-after-extensions` to `vmss extension set` to define what extensions must be provisioned before the extension being set
- Added argument `--replica-count` to `sig image-version update` for setting the default replication count
- Fixed bug with `image create --source` where source os disk is mistaken for a VM with the same name, even if the full resource ID is provided

December 20, 2018

Version 2.0.54

Appservice

- Fixed issue where `webapp up` would fail to redeploy
- Added support for listing and restoring webapp snapshots
- Added support for `--runtime` flag to Windows function apps

IoTCentral

- Fixed update command API call

Role

- [BREAKING CHANGE] Changed `ad [app|sp] list` to only list the first 100 objects by default

SQL

- Added support for custom collation on managed instances

VM

- Added `--os-type` parameter to `disk create`

December 18, 2018

Version 2.0.53

ACR

- Added support for image import from external Container Registries
- Condensed the table layout for task list
- Added support for Azure DevOps URLs

ACS

- Added Virtual Nodes Preview
- Removed "(PREVIEW)" from AAD arguments to `aks create`
- [DEPRECATED] Deprecated `az acs` commands. The ACS service will retire on January 31, 2020
- Added support of Network Policy when creating new AKS clusters
- Removed requirement of `--nodepool-name` argument for `aks scale` if there's only one nodepool

Appservice

- Fixed issue where `webapp config container` did not honor `--slot` parameter

Botservice

- Added support for `.bot` file parsing when calling `bot show`
- Fixed AppInsights provisioning bug
- Fixed whitespace bug when dealing with file paths
- Reduced Kudu network calls
- General command UX improvements

Consumption

- Fixed bugs for budget API to show notifications

CosmosDB

- Added support for updating account from multi-master to single-master

Maps

- Added support for the S1 SKU to `maps account [create|update]`

Network

- Added support for `--format` and `--log-version` to `watcher flow-log configure`
- Fixed issue with `dns zone update` where using "" to clear resolution and registration VNets didn't work

Resource

- Fixed handling of scope parameter for management groups in `policy assignment [create|list|delete|show|update]`
- Added new command `resource wait`

Storage

- Added ability to update log schema version for storage services in `storage logging update`

VM

- Fixed crash in `vm identity remove` when the specified vm has no assigned managed service identities

December 4, 2018

Version 2.0.52

Core

- Added support for cross tenant resource provisioning for multi-tenant service principal
- Fixed bug where ids piped from a command with tsv output was improperly parsed

Appservice

- [PREVIEW] Added `webapp up` command that helps in creating & deploying contents to app
- Fixed a bug on container based windows app due to backend change

Network

- Added `--exclusion` argument to `application-gateway waf-config set` to support WAF exclusions

Role

- Added support for custom identifiers for password credential

VM

- [DEPRECATED] Deprecated `vm extension [show|wait] --expand` parameter
- Added `--force` parameter to `vm restart` to redeploy unresponsive VMs
- Changed `[vm|vmss] create --authentication-type` to accept "all" to create a VM with both password and ssh authentication
- Added `image create --os-disk-caching` parameter to set os disk caching for an image

November 20, 2018

Version 2.0.51

Core

- Changed MSI login to not reuse subscription name in identity

ACR

- Added context token to task step
- Added support for setting secrets in acr run to mirror acr task
- Improved support for `--top` and `--orderby` for `show-tags` and `show-manifests` commands

Appservice

- Changed zip deployment default timeout to poll for the status increased to 5 mins, also adding a timeout property to customize this value
- Updated the default `node_version`. Resetting slot swap action, during a two phase swap preserves all the appsettings & connection strings
- Removed client-side SKU check for Linux app service plan create
- Fixed error when trying to get zipdeploy status

IoTCentral

- Added subdomain availability check when creating an IoT Central application

KeyVault

- Fixed bug where errors may have been ignored

Network

- Added `root-cert` subcommands to `application-gateway` to handle trusted root certificates
- Added `--min-capacity` and `--custom-error-pages` options to `application-gateway [create|update]`:
- Added `--zones` for availability zone support to `application-gateway create`

- Added arguments `--file-upload-limit`, `--max-request-body-size` and `--request-body-check` to `application-gateway waf-config set`

Rdbms

- Added mariadb vnet commands

Rbac

- Fixed an issue with attempting to update immutable credentials in `ad app update`
- Added output warnings to communicate breaking changes in the near future for `ad [app|sp] list`

Storage

- Improved handling of corner cases for storage copy commands
- Fixed issue with `storage blob copy start-batch` not using login credentials when the destination and source accounts are the same
- Fixed bug with `storage [blob|file] url` where `sas_token` wasn't incorporated into URL
- Added breaking change warning to `[blob|container] list`: will soon output only first 5000 results by default

VM

- Added support to `[vm|vmss] create --storage-sku` to specify the storage account SKU for managed OS and data disks separately
- Changed version name parameters to `sig image-version` to be `--image-version -e`
- Deprecated `sig image-version` argument `--image-version-name`, replaced by `--image-version`
- Added support to use local OS disk to `[vm|vmss] create --ephemeral-os-disk`
- Added support for `--no-wait` to `snapshot create/update`
- Added `snapshot wait` command
- Added support for using instance name with `[vm|vmss] extension set --extension-instance-name`

November 6, 2018

Core

- Added support for service principal sn+issuer auth

ACR

- Added support for commit and pull request git events for Task source trigger
- Changed to use default Dockerfile if it's not specified in build command

ACS

- [BREAKING CHANGE] Removed `enable_cloud_console_aks_browse` to enable 'az aks browse' by default

Advisor

- GA release

AMS

- Added new command groups:
 - `ams account-filter`
 - `ams asset-filter`
 - `ams content-key-policy`
 - `ams live-event`
 - `ams live-output`
 - `ams streaming-endpoint`
 - `ams mru`
- Added new commands:
 - `ams account check-name`
 - `ams job update`
 - `ams asset get-encryption-key`
 - `ams asset get-streaming-locators`
 - `ams streaming-locator get-content-keys`
- Added encryption parameters support to `ams streaming-policy create`
- Added support to `ams transform output remove` now can be performed by passing the output index to remove

- Added `--correlation-data` and `--label` arguments to `ams job` command group
- Added `--storage-account` and `--container` arguments to `ams asset` command group
- Added default values for expiry time (Now+23h) and permissions (Read) in `ams asset get-sas-url` command
- [BREAKING CHANGE] Replaced `ams streaming locator` command with `ams streaming-locator`
- [BREAKING CHANGE] Updated `--content-keys` argument of `ams streaming locator`
- [BREAKING CHANGE] Renamed `--content-policy-name` to `--content-key-policy-name` in `ams streaming locator` command
- [BREAKING CHANGE] Replaced `ams streaming policy` command with `ams streaming-policy`
- [BREAKING CHANGE] Replaced `--preset-names` argument with `--preset` in `ams transform` command group. Now you can only set 1 output/preset at a time (to add more you have to run `ams transform output add`). Also, you can set custom StandardEncoderPreset by passing the path to your custom JSON
- [BREAKING CHANGE] Renamed `--output-asset-names` to `--output-assets` in `ams job start` command. Now it accepts a space-separated list of assets in 'assetName=label' format. An asset without label can be sent like this: 'assetName='

AppService

- Fixed a bug in `az webapp config backup update` that prevents setting a backup schedule if one is not already set

Configure

- Added YAML to output format options

Container

- Changed to show identity when exporting a container group to yaml

EventHub

- Added `--enable-kafka` flag to support Kafka in `eventhub namespace` [`create|update`]

Interactive

- Interactive now installs the `interactive` extension, which will allow for faster updates and support

Monitor

- Added support for metric names which include characters forward-slash (/) and period (.) to `--condition` in `monitor metrics alert [create|update]`

Network

- Deprecated `network interface-endpoint` command names in favor of `network private-endpoint`
- Fixed issue with where `--peer-circuit` argument in `express-route peering connection create` would not accept an ID
- Fixed issue where `--ip-tags` did not work correctly with `public-ip create`

Profile

- Added `--use-cert-sn-issuer` to `az login` for service principal login with cert auto-rolls

RDBMS

- Added mysql replica commands

Resource

- Added support for management groups and subscriptions to `policy definition|set-definition` commands

Role

- Added support for API permission management, signed-in-user, and application password & certificate credential management
- Changed `ad sp create-for-rbac` to clarify the confusion between `displayName` and service principal name
- Added support to grant permissions to AAD apps

Storage

- Added support to connect to storage services only with SAS and endpoints (without an account name or a key) as described in [Configure Azure Storage connection strings](#).

VM

- Added `storage-sku` argument to `image create` for setting the image's default storage account type
- Fixed bug with `vm resize` where `--no-wait` option causes command to crash
- Changed `vm encryption show` table output format to show status
- Changed `vm secret format` to require json/jsonc output. Warns user and defaults to json output if an undesired output format is selected
- Improved argument validation for `vm create --image`

October 23, 2018

Version 2.0.49

Core

- Fixed issue with `--ids` where `--subscription` would take precedence over the subscription in `--ids`
- Added explicit warnings when parameters would be ignored by use of `--ids`

ACR

- Fixed an ACR Build encoding issue in Python2

CDN

- [BREAKING CHANGE] Changed `cdn endpoint create`'s default query string caching behaviour to no longer defaults to "IgnoreQueryString". It is now set by the service

Container

- Added `Private` as a valid type to pass to '--ip-address'

- Changed to allow using only subnet ID to setup a virtual network for the container group
- Changed to allow using vnet name or resource id to enable using vnets from different resource groups
- Added `--assign-identity` for adding a MSI identity to a container group
- Added `--scope` to create a role assignment for the system assigned MSI identity
- Added a warning when creating a container group with an image without a long running process
- Fixed table output issues for `list` and `show` commands

CosmosDB

- Added `--enable-multiple-write-locations` support to `cosmosdb create`

Interactive

- Changed to ensure global subscription parameter appears in parameters

IoT Central

- Added template and display name options for IoT Central Application creation
- [BREAKING CHANGE] Removed support for the F1 SKU; Use S1 SKU instead

Monitor

- Changes to `monitor activity-log list`:
 - Added support for listing all events at the subscription level
 - Added `--offset` parameter to more easily create time queries
 - Improved validation for `--start-time` and `--end-time` to use wider range of ISO8601 formats and more user-friendly datetime formats
 - Added `-namespace` as alias for deprecated option `-resource-provider`
 - Deprecated `--filters` because no values other than those with strongly-typed options are supported by the service
- Changes to `monitor metrics list`:
 - Added `--offset` parameter to more easily create time queries
 - Improved validation for `--start-time` and `--end-time` to use wider range of ISO8601 formats and more user-friendly datetime formats
- Improved validation for `--event-hub` and `--event-hub-rule` arguments to `monitor diagnostic-settings create`

Network

- Added `--app-gateway-address-pools` and `--gateway-name` arguments to `nic create`, to support adding application gateway backend address pools to a NIC
- Added `--app-gateway-address-pools` and `--gateway-name` arguments to `nic ip-config create/update`, to support adding application gateway backend address pools to a NIC

ServiceBus

- Added Read-Only `migration_state` to `MigrationConfigProperties` to show current Service Bus Standard to Premium namespace migration state

SQL

- Fixed `sql failover-group create` and `sql failover-group update` to work with Manual failover policy

Storage

- Fixed `az storage cors list` output formatting, all items show correct "Service" key
- Added `--bypass-immutability-policy` parameter for immutability-policy blocked container deletion

VM

- Enforce disk caching mode be `None` for Lv/Lv2 series of machines in `[vm|vmss] create`
- Updated supported size list supporting networking accelerator for `vm create`
- Added strong typed arguments for ultrassd iops and mbps configs for `disk create`

October 16, 2018

Version 2.0.48

VM

- Fixed SDK issue that caused Homebrew instllation to fail

October 9, 2018

Version 2.0.47

Core

- Improved error handling for "Bad Request" errors

ACR

- Added support for similar table format as helm client

ACS

- Added `aks [create|scale] --nodepool-name` to configure nodepool name, truncated to 12 characters, default - nodepool1
- Fixed to fall back to 'scp' when Parimiko fails
- Changed `aks create` to no longer require `--aad-tenant-id`
- Improved merging of Kubernetes credentials when duplicate entries are present

Container

- Changed `functionapp create` to support creating a Linux consumption plan type with a specific runtime
- [PREVIEW] Added support for hosting webapps on Windows containers

Event Hub

- Fixed `eventhub update` command
- [BREAKING CHANGE] Changed `list` commands to handle errors for resource(s) `NotFound(404)` in the typical way instead of showing empty list

Extensions

- Fixed issue with attempting to add an extension that is already installed

HDInsight

- Initial release

IoT

- Added extension installation command to first-run banner

KeyVault

- Changed to restrict keyvault storage commands to the latest API profile

Network

- Fixed `network dns zone create`: Command succeeds even if the user has configured a default location. See #6052
- Deprecated `--remote-vnet-id` for `network vnet peering create`
- Added `--remote-vnet` to `network vnet peering create` which accepts a name or ID
- Added support for multiple subnet prefixes to `network vnet create` with `--subnet-prefixes`
- Added support for multiple subnet prefixes to `network vnet subnet [create|update]` with `--address-prefixes`
- Fixed issue with `network application-gateway create` that prevented creating gateways with `WAF_v2` or `Standard_v2` SKU
- Added `--service-endpoint-policy` convenience argument to `network vnet subnet update`

Role

- Added support for listing Azure AD app owners to `ad app owner`
- Added support for listing Azure AD service principal owners to `ad sp owner`
- Changed to ensure role definition create & update commands accept multiple permission configurations
- Changed `ad sp create-for-rbac` to ensure home page URI is always "https"

Service Bus

- [BREAKING CHANGE] Changed `list` commands to handle errors for resource(s) `NotFound(404)` in the typical way instead of showing empty list

VM

- Fixed empty `accessSas` field in `disk grant-access`

- Changed `vmss create` to reserve large enough frontend port range to handle overprovisioning
- Fixed update commands for `sig`
- Added `--no-wait` support for managing image versions in `sig`
- Changed `vm list-ip-addresses` to show availability zone of public IP addresses
- Changed `[vm|vmss] disk attach` to set disk's default lun to the first available spot

September 21, 2018

Version 2.0.46

ACR

- Added ACR Task commands
- Added quick run command
- Deprecated `build-task` command group
- Added `helm` command group to support managing helm charts with ACR
- Added support for idempotent create for managed registry
- Added a no-format flag for displaying build logs

ACS

- Changed the `install-connector` command to set the AKS Master FQDN
- Fixed creating role assignment for vnet-subnet-id when not specifying service principal and skip-role-assigemnt

AppService

- Added support for webjobs (continuous and triggered) operations management
- az webapp config set supports `--fts-state` propertyAlso added support fot az functionapp config set & show
- Added support for bring your own storage for webapps
- Added support for listing and restoring deleted webapps

Batch

- Changed adding tasks through `--json-file` to support `AddTaskCollectionParameter` syntax
- Updated documentation of accepted `--json-file` formats

- Added `--max-tasks-per-node-option` to `batch pool create`
- Changed behavior of `batch account` to show currently logged in account if no options are specified

Batch AI

- Fixed auto storage account creation failure in `batchai cluster create` command

Cognitive Services

- Added completer for `--sku`, `--kind`, `--location` arguments
- Added command `cognitiveservices account list-usage`
- Added command `cognitiveservices account list-kinds`
- Added command `cognitiveservices account list`
- Deprecated `cognitiveservices list`
- Changed `--name` to be optional for `cognitiveservices account list-skus`

Container

- Added ability to restart and stop a running container group
- Added `--network-profile` for passing in a network profile
- Added `--subnet`, `--vnet_name`, to allow creating container groups in a VNET
- Changed table output to show the status of the container group

Datalake

- Added commands for virtual network rules

Interactive Shell

- Fixed error on Windows where commands fail to run properly
- Fixed command loading problem in interactive that was caused by deprecated objects

IoT

- Added support for routing IoT Hubs

Key Vault

- Fixed Key Vault key import for RSA keys

Network

- Add `network public-ip prefix` commands to support public IP prefixes features
- Add `network service-endpoint` commands to support service endpoint policy features
- Add `network lb outbound-rule` commands to support creation of Standard Load Balancer outbound rules
- Add `--public-ip-prefix` to `network lb frontend-ip create/update` to support frontend IP configurations using public IP prefixes
- Add `--enable-tcp-reset` to `network lb rule/inbound-nat-rule/inbound-nat-pool create/update`
- Add `--disable-outbound-snat` to `network lb rule create/update`
- Allow `network watcher flow-log show/configure` to be used with classic NSGs
- Add `network watcher run-configuration-diagnostic` command
- Fix `network watcher test-connectivity` command and add `--method`, `--valid-status-codes` and `--headers` properties
- `network express-route create/update`: Add `--allow-global-reach` flag
- `network vnet subnet create/update`: Add support for `--delegation`
- Added `network vnet subnet list-available-delegations` command
- `network traffic-manager profile create/update`: Added support for `--interval`, `--timeout` and `--max-failures` for Monitor configuration Deprecated options `--monitor-path`, `--monitor-port` and `--monitor-protocol` in favor of `--path`, `--port`, `--protocol`
- `network lb frontend-ip create/update`: Fixed the logic for setting private IP allocation methodIf a private IP address is provided, the allocation will be staticIf no private IP address is provided, or empty string is provided for private IP address, allocation is dynamic.
- `dns record-set * create/update`: Add support for `--target-resource`
- Add `network interface-endpoint` commands to query interface endpoint objects
- Add `network profile show/list/delete` for partial management of network profiles
- Add `network express-route peering connection` commands to manage peering connections between ExpressRoutes

RDBMS

- Added support for MariaDB service

Reservation

- Added CosmosDb in the reserved resource enum type
- Added name property in Patch model

Manage App

- Fixed bug in `managedapp create --kind MarketPlace` causing instance creation of a Marketplace managed app to crash
- Changed `feature` commands to be restricted to supported profiles

Role

- Added support for listing user's group memberships

SignalR

- First release

Storage

- Added `--auth-mode login` parameter for use of user's login credentials for blob and queue authorization
- Added `storage container immutability-policy/legal-hold` to manage immutable storage

VM

- Fixed issue where `vm create --generate-ssh-keys` overwrites private key file if public key file is missing (#4725, #6780)
- Added support for shared image gallery through `az sig`

August 28, 2018

Version 2.0.45

Core

- Fixed issue of loading empty configuration file

- Added support for profile `2018-03-01-hybrid` for Azure Stack

ACR

- Added a workaround for runtime operations without ARM requests
- Changed to exclude version control files (eg, `.git`, `.gitignore`) from uploaded tar by default in `build` command

ACS

- Changed `aks create` to defaults to `Standard_DS2_v2` VMs
- Changed `aks get-credentials` to now call new apis to get cluster credential

AppService

- Added support for CORS on functionapp & webapp
- Added ARM tag support on create commands
- Changed `[webapp|functionapp] identity show` to exit with code 3 upon a missing resource

Backup

- Changed `backup vault backup-properties show` to exit with code 3 upon a missing resource

Bot Service

- Initial Bot Service CLI Release

Cognitive Services

- Added new parameter `--api-properties`, which is required for creating some of the services

IoT

- Fixed issue with associating linked hubs

Monitor

- Added `monitor metrics alert` commands for near-realtime metric alerts
- Deprecated `monitor alert` commands

Network

- Changed `network application-gateway ssl-policy predefined show` to exit with code 3 upon a missing resource

Resource

- Changed `provider operation show` to exit with code 3 upon a missing resource

Storage

- Changed `storage share policy show` to exit with code 3 upon a missing resource

VM

- Changed `vm/vmss identity show` to exit with code 3 upon a missing resource
- Deprecated `--storage-caching` for `vm create`

August 14, 2018

Version 2.0.44

Core

- Fixed numeric display in `table` output
- Added YAML output format

Telemetry

- Improved telemetry reporting

ACR

- Added `content-trust policy` commands
- Fixed issue where `.dockerignore` was not handled properly

ACS

- Changed `az acs/aks install-cli` to install under `%USERPROFILE%\azure-kubectl` on Windows
- Changed `az aks install-connector` to detect if the cluster has RBAC and configure ACI Connector appropriately
- Changed to role assignment to the subnet when it's provided
- Added new option to "skip role assignment" for subnet when it's provided
- Changed to skip role assignment to subnet when assignment already exists

AppService

- Fixed a bug that prevent from creating a function-app using storage accounts in external resource groups
- Fixed a crash on zip deployment

BatchAI

- Changed logger output for auto-storage account creation to specifies "resource group".

Container

- Added `--secure-environment-variables` for passing secure environment variables to a container

IoT

- [BREAKING CHANGE] Removed deprecated commands which have moved to the iot extension
- Updated elements to not assume `azure-devices.net` domain

IoT Central

- Initial release of IoT Central module

KeyVault

- Added commands for managing storage accounts and sas-definitions
- Added commands for network-rules

- Added `--id` parameter to secret, key, and certificate operations
- Added support for KV mgmt multi-api version
- Added support for KV data plane multi-api version

Relay

- Initial release

Sql

- Added `sql failover-group` commands

Storage

- [BREAKING CHANGE] Changed `storage account show-usage` to require `--location` parameter and will list by region
- Changed `--resource-group` parameter to be optional for `storage account` commands
- Removed 'Failed precondition' warnings for individual failures in batch commands for single aggregated message
- Changed `[blob|file] delete-batch` commands to no longer output array of nulls
- Changed `blob [download|upload|delete-batch]` commands to read sas-token from container url

VM

- Added common filters to `vm list-skus` for ease of use

July 31, 2018

Version 2.0.43

ACR

- Added `--with-secure-properties` flag to `acr build-task show` command
- Added `acr build-task update-build` command

ACS

- Changed to return 0 (success) when ending `az aks browse` by pressing [Ctrl+C]

Batch

- Fixed bug when showing AAD token in cloudshell

Container

- Removed requirement for `--log-analytics-workspace-key` for name or ID when in set subscription

Network

- Added dns support to 2017-03-09-profile for Azure Stack

Resource

- Added `--rollback-on-error` to `group deployment create` to execute a known-good deployment on error
- Fixed issue where `--parameters {}` with `group deployment create` resulted in an error

Role

- Added support for stack profile 2017-03-09-profile
- Fixed issue where generic update parameters to `app update` would not work correctly

Search

- Added commands for Azure Search service

Service Bus

- Added migration command group to migrate a namespace from Service Bus Standard to Premium
- Added new optional properties to Service Bus queue and Subscription

- `--enable-batched-operations` and `--enable-dead-lettering-on-message-expiration` in `queue`
- `--dead-letter-on-filter-exceptions` in `subscriptions`

Storage

- Added support for download of large files using a single connection
- Converted `show` commands that were missed from failing with exit code 3 upon a missing resource

VM

- Added support to list availability sets by subscription
- Added support for `StandardSSD_LRS`
- Added support for application security group on creating VM scale set
- [BREAKING CHANGE] Changed `[vm|vmss] create`, `[vm|vmss] identity assign`, and `[vm|vmss] identity remove` to output user assigned identities in dictionary format

July 18, 2018

Version 2.0.42

Core

- Added support for browser-based login in WSL bash window
- Added `--force-string` flag to all generic update commands
- [BREAKING CHANGE] Changed 'show' commands to log error message and fail with exit code of 3 upon a missing resource

ACR

- [BREAKING CHANGE] Updated '--no-push' to a pure flag in 'acr build' command
- Added `show` and `update` commands under `acr repository` group
- Added `--detail` flag for `show-manifests` and `show-tags` to show more detailed information
- Added `--image` parameter to support get build details or logs by an image

ACS

- Changed `az aks create` to error out if `--max-pods` is less than 5

AppService

- Added support for PremiumV2 skus

Batch

- Fixed bug on using token credential on cloud shell mode
- Changed JSON input to be case-insensitive

Batch AI

- Fixed `az batchai job exec` command

Container

- Removed the requirement for username and password for non dockerhub registries
- Fixed error when creating container groups from yaml file

Network

- Added `--no-wait` support to `network nic [create|update|delete]`
- Added `network nic wait`
- Deprecated `--ids` argument for `network vnet [subnet|peering] list`
- Added `--include-default` flag to include default security rules in the output of `network nsg rule list`

Resource

- Added `--no-wait` support to `group deployment delete`
- Added `--no-wait` support to `deployment delete`
- Added `deployment wait` command
- Fixed issue where the subscription-level `az deployment` commands erroneously appeared for profile 2017-03-09-profile

SQL

- Fixed 'The provided resource group name ... did not match the name in the Url' error when specifying elastic pool name for `sql db copy` and `sql db replica create` commands
- Allow configuring default sql server by executing `az configure --defaults sql-server=<name>`
- Implemented table formatters for `sql server`, `sql server firewall-rule`, `sql list-usages`, and `sql show-usage` commands

Storage

- Added `pageRanges` property to `storage blob show` output that will be populated for page blobs

VM

- [BREAKING CHANGE] Changed `vmss create` to use `Standard_DS1_v2` as the default instance size
- Added `--no-wait` support to `vm extension [set|delete]` and `vmss extension [set|delete]`
- Added `vm extension wait`

July 3, 2018

Version 2.0.41

AKS

- Changed monitoring to use subscription ID

Version 2.0.40

Core

- Added a new authorization code flow for interactive login

ACR

- Added polling build status

- Added support for case-insensitive enum values
- Added `--top` and `--orderby` parameters for `show-manifests`

ACS

- [BREAKING CHANGE] Enable Kubernetes role-based access control by default
- Added `--disable-rbac` argument and deprecated `--enable-rbac` since it's the default now
- Updated options for `aks browse` command. Added `--listen-port` support
- Updated the default helm chart package for `aks install-connector` command. Use `virtual-kubelet-for-aks-latest.tgz`
- Added `aks enable-addons` and `aks disable-addons` commands to update an existing cluster

AppService

- Added support for disabling identity via `webapp identity remove`
- Removed `preview` tag for Identity feature

Backup

- Updated module definition

BatchAI

- Fixed table output for `batchai cluster node list` and `batchai job node list` commands

Cloud

- Added `acr login` server suffix to cloud config

Container

- Changed `container create` to default to long running operation
- Added Log Analytics parameters `--log-analytics-workspace` and `--log-analytics-workspace-key`
- Added `--protocol` parameter to specify which network protocol to use

Extension

- Changed `extension list-available` to only show extensions compatible with CLI version

Network

- Fixed issue where record types were case-sensitive ([#6602](#))

Rdbms

- Added `[postgres|mysql] server vnet-rule` commands

Resource

- Added new operation group `deployment`

VM

- Added support for removing system assigned identity

June 25, 2018

Version 2.0.39

CLI

- Updated file trimming in MSI installer to fix extension installation issue

June 19, 2018

Version 2.0.38

Core

- Added global support for `--subscription` to most commands

ACR

- Added `azure-storage-blob` as dependency
- Changed default CPU configuration with `acr build-task create` to use 2 cores

ACS

- Updated options of `aks use-dev-spaces` command. Added `--update` support
- Changed `aks get-credentials --admin` to not replace the user context in `$HOME/.kube/config`
- Exposed read-only `nodeResourceGroup` property on managed clusters
- Fixed `acs browse` command error
- Made `--connector-name` optional for `aks install-connector`, `aks upgrade-connector` and `aks remove-connector`
- Added new Azure Container Instance regions for `aks install-connector`
- Added the normalized location into the helm release name and node name to `aks install-connector`

AppService

- Added support for newer versions of `urllib`
- Added support to `functionapp create` to use appservice plan from external resource groups

Batch

- Removed `azure-batch-extensions` dependency

Batch AI

- Added support for workspaces. Workspaces allow to group clusters, file-servers and experiments in groups removing limitation on number of resources can be created
- Added support for experiments. Experiments allow to group jobs in collections removing limitation on number of created jobs
- Added support to configure `/dev/shm` for jobs running in a docker container
- Added `batchai cluster node exec` and `batchai job node exec` commands. These commands allow to execute any commands directly on nodes and provide functionality for port forwarding.
- Added support for `--ids` to `batchai` commands

- [BREAKING CHANGE] All clusters and fileservers must be created under workspaces
- [BREAKING CHANGE] Jobs must be created under experiments
- [BREAKING CHANGE] Removed `--nfs-resource-group` from `cluster create` and `job create` commands. To mount an NFS belonging to a different workspace/resource group provide file server's ARM ID via `--nfs` option
- [BREAKING CHANGE] Removed `--cluster-resource-group` from `job create` command. To submit a job on a cluster belonging to a different workspace/resource group provide cluster's ARM ID via `--cluster` option
- [BREAKING CHANGE] Removed `location` attribute from jobs, cluster and file servers. Location now is an attribute of a workspace.
- [BREAKING CHANGE] Removed `--location` from `job create`, `cluster create` and `file-server create` commands
- [BREAKING CHANGE] Changed names of short options to make interface more consistent:
 - Renamed `[--config, -c]` to `[--config-file, -f]`
 - Renamed `[--cluster, -r]` to `[--cluster, -c]`
 - Renamed `[--cluster, -n]` to `[--cluster, -c]`
 - Renamed `[--job, -n]` to `[--job, -j]`

Maps

- [BREAKING CHANGE] Changed `maps account create` to require accepting Terms of Service either by interactive prompt or `--accept-tos` flag

Network

- Added support for `https` to `network 1b probe create` #6571 ↗
- Fixed issue where `--endpoint-status` was case sensitive. #6502 ↗

Reservations

- [BREAKING CHANGE] Added required parameter `ReservedResourceType` to `reservations catalog show`
- Added parameter `Location` to `reservations catalog show`
- [BREAKING CHANGE] Removed `kind` from `ReservationProperties`
- [BREAKING CHANGE] Renamed `capabilities` to `sku_properties` in `Catalog`
- [BREAKING CHANGE] Removed `size` and `tier` properties from `Catalog`

- Added parameter `InstanceFlexibility` to `reservations reservation update`

Role

- Improved error handling

SQL

- Fixed confusing error when running `az sql db list-editions` for a location that is not available to your subscription

Storage

- Changed table output for `storage blob download` to be more readable

VM

- Improved refine vm size check for accelerated networking support in `vm create`
- Added warning for `vmss create` that the default vm size will be switched from `Standard_D1_v2` to `Standard_DS1_v2`
- Added `--force-update` to `[vm|vmss] extension set` to update the extension even when the configuration has not changed

June 13, 2018

Version 2.0.37

Core

- Improved interactive telemetry

Version 2.0.36

AKS

- Added advanced networking options to `aks create`
- Added arguments to `aks create` to enable monitoring and HTTP routing
- Added `--no-ssh-key` argument to `aks create`

- Added `--enable-rbac` argument to `aks create`
- [PREVIEW] Added support for Azure Active Directory authentication to `aks create`

AppService

- Fixed an issue with incompatible urllib versions

June 5, 2018

Version 2.0.35

Interactive

- Added limits to the dependencies of interactive mode

Version 2.0.34

Core

- Added support for cross tenant resource referencing
- Improved telemetry upload reliability

ACR

- Added support for VSTS as a remote source location
- Added `acr import` command

AKS

- Changed `aks get-credentials` to create the kube config file with more secure filesystem permissions

Batch

- Fixed bug in Pool list table formatting [[Issue #4378 ↗](#)]

IOT

- Added support for creating Basic Tier IoT Hubs

Network

- Improved `network vnet peering`

Policy Insights

- Initial Release

ARM

- Added `account management-group` commands.

SQL

- Added new managed instance commands:
 - `sql mi create`
 - `sql mi show`
 - `sql mi list`
 - `sql mi update`
 - `sql mi delete`
- Added new managed database commands:
 - `sql midb create`
 - `sql midb show`
 - `sql midb list`
 - `sql midb restore`
 - `sql midb delete`

Storage

- Added extra mimetypes for json and javascript to be inferred from file extensions

VM

- Changed `vm list-skus` to use fixed columns and add warning that `Tier` and `Size` will be removed
- Added `--accelerated-networking` option to `vm create`
- Added `--tags` to `identity create`

May 22, 2018

Version 2.0.33

Core

- Added support for expanding @ in file names

ACS

- Added new Dev-Spaces commands `aks use-dev-spaces` and `aks remove-dev-spaces`
- Fixed typo in help message

AppService

- Improved generic update commands
- Added async support for `webapp deployment source config-zip`

Container

- Added support for exporting a container group in yaml format
- Added support for using a yaml file to create / update a container group

Extension

- Improved removal of extensions

Interactive

- Changed logging to mute parser for completions
- Improved handling of bad help caches

KeyVault

- Fixed keyvault commands to work in cloud shell or VMs with identity

Network

- Fix issue where `network watcher show-topology` would not work with vnet and/or subnet name [#6326](#)
- Fix issue where some `network watcher` commands would claim Network Watcher is not enabled for regions when it actually is [#6264](#)

SQL

- [BREAKING CHANGE] Changed response objects returned from `db` and `dw` commands:
 - Renamed `serviceLevelObjective` property to `currentServiceObjectiveName`
 - Removed `currentServiceObjectiveId` and `requestedServiceObjectiveId` properties
 - Changed `maxSizeBytes` property to be an integer value instead of a string
- [BREAKING CHANGE] Changed the following `db` and `dw` properties to be read-only:
 - `requestedServiceObjectiveName`. To update, use the `--service-objective` parameter or set the `sku.name` property
 - `edition`. To update, use the `--edition` parameter or set the `sku.tier` property
 - `elasticPoolName`. To update, use the `--elastic-pool` parameter or set the `elasticPoolId` property
- [BREAKING CHANGE] Changed the following `elastic-pool` properties to be read-only:
 - `edition`. To update, use the `--edition` parameter
 - `dtu`. To update, use the `--capacity` parameter
 - `databaseDtuMin`. To update, use the `--db-min-capacity` parameter
 - `databaseDtuMax`. To update, use the `--db-max-capacity` parameter
- Added `--family` and `--capacity` parameters to `db`, `dw`, and `elastic-pool` commands.
- Added table formatters to `db`, `dw`, and `elastic-pool` commands.

Storage

- Added completer for `--account-name` argument
- Fixed problem with `storage entity query`

VM

- [BREAKING CHANGE] Removed `--write-accelerator` from `vm create`. The same support can be accessed through `vm update` or `vm disk attach`
- Fixed extension image matching in `[vm|vmss]` extension
- Added `--boot-diagnostics-storage` to `vm create` to capture boot log
- Added `--license-type` to `[vm|vmss]` update

May 7, 2018

Version 2.0.32

Core

- Fixed an unhandled exception when retrieving secrets from a service principal account with cert
- Added limited support for positional arguments
- Fix issue where `--query` could not be used with `--ids`. [#5591 ↗](#)
- Improved piping scenarios from commands when using `--ids`. Supports `-o tsv` with a query specified or `-o json` without specifying a query
- Added command suggestions on error if users have typo in their commands
- Improved error when users type `az ''`
- Added support custom resource types for command modules and extensions

ACR

- Added ACR Build commands
- Improved resource not found error messages
- Improved resource creation performance and error handling
- Improved acr login in non-standard consoles and WSL
- Improved repository commands error messages
- Updated table columns and ordering

ACS

- Added warning that `az aks` is a preview service
- Fixed the permission issue in `aks install-connector` when `--aci-resource-group` is not specified

AMS

- Initial release - Manage Azure Media Services resources

Appservice

- Fixed a bug in `webapp delete` when `--slot` is provided
- Removed `--runtime-version` from `webapp auth update`
- Added support for `min_tls_version` & `https2.0`
- Added support for multicontainers

Batch AI

- Changed `batchai create cluster` to respect vm priority configured in the cluster's configuration file

Cognitive Services

- Fixed typo in example for `cognitiveservices account create` #5603 ↗

Consumption

- Added new commands for budget API

Container

- Removed requirement for `--registry-server` for `container create` when a registry server is included in the image name

Cosmos DB

- Introducing VNET support for Azure CLI - Cosmos DB

DMS

- Initial release - Adds support for the SQL to Azure SQL migration scenario

Extension

- Fixed bug where extension metadata stopped being shown

Interactive

- Allow interactive completers to function with positional arguments
- More user-friendly output when users type ''
- Fixed completions for parameters with no help
- Fixed descriptions for command-groups

Lab

- Fixed regressions from knack conversion

Network

- [BREAKING CHANGE] Removed the `--ids` parameter for:
 - `express-route auth list`
 - `express-route peering list`
 - `nic ip-config list`
 - `nsg rule list`
 - `route-filter rule list`
 - `route-table route list`
 - `traffic-manager endpoint list`

Profile

- Fixed `disk create` source detection
- [BREAKING CHANGE] Removed `--msi-port` and `--identity-port` as they are no longer used
- Fixed typo in `account get-access-token` short summary

Redis

- Deprecated `redis patch-schedule patch-schedule show` in favor of `redis patch-schedule show`
- Deprecated `redis list-all`. This functionality has been folded into `redis list`
- Deprecated `redis import-method` in favor of `redis import`
- Added support for `--ids` to various commands

Role

- [BREAKING CHANGE] Removed deprecated `ad sp reset-credentials`

Storage

- Allow destination sas-token to apply to source for blob copy if source sas and account key are unspecified
- Exposed --socket-timeout for blob uploads and downloads
- Treat blob names that start with path separators as relative paths
- Allow `storage blob copy --source-sas` with starting query char, '?'
- Fixed `storage entity query --marker` to accept list of key=values

VM

- Fixed an invalid detection logic on unmanaged blob uri
- Added support disk encryption w/o user provided service principals
- [BREAKING CHANGE] Do not use VM 'ManagedIdentityExtension' for MSI support
- Added support for eviction policy to `vmss`
- [BREAKING CHANGE] Removed `--ids` from:
 - `vm extension list`
 - `vm secret list`
 - `vm unmanaged-disk list`
 - `vmss nic list`
- Added write accelerator support
- Added `vmss perform-maintenance`
- Fixed `vm diagnostics set` to detect VM's OS type reliably
- Changed `vm resize` to check if the requested size is different than currently set and update only on change

April 10, 2018

Version 2.0.31

ACR

- Improved error handling of wincred fallback

ACS

- Changed aks created SPNs to be valid for 5 years

Appservice

- [BREAKING CHANGE]: Removed `assign-identity`
- Fixed uncaught exception for nonexistent webapp plans

BatchAI

- Added support for 2018-03-01 API
 - Job level mounting
 - Environment variables with secret values
 - Performance counters settings
 - Reporting of job specific path segment
 - Support for subfolders in list files api
 - Usage and limits reporting
 - Allow to specify caching type for NFS servers
 - Support for custom images
 - Added pyTorch toolkit support
- Added `job wait` command which allows to wait for the job completion and reports job exit code
- Added `usage show` command to list current Batch AI resources usage and limits for different regions
- National clouds are supported
- Added job command line arguments to mount filesystems on the job level in addition to config files
- Added more options to customize clusters - vm priority, subnet, initial nodes count for auto-scale clusters, specifying custom image
- Added command line option to specify caching type for Batch AI managed NFS
- Simplified specifying mount filesystem in config files. Now you can omit credentials for Azure File Share and Azure Blob Containers - CLI will populate missing credentials using storage account key provided via command line parameters or specified via environment variable or will query the key from Azure Storage (if the storage account belongs to the current subscription)
- Job file stream command now auto-completes when the job is completed (succeeded, failed, terminated or deleted)
- Improved `table` output for `show` operations

- Added `--use-auto-storage` option for cluster creation. This option make it simpler to manage storage accounts and mount Azure File Share and Azure Blob Containers to clusters
- Added `--generate-ssh-keys` option to `cluster create` and `file-server create`
- Added ability to provide node setup task via command line
- [BREAKING CHANGE] Moved `job stream-file` and `job list-files` commands under `job file` group
- [BREAKING CHANGE] Renamed `--admin-user-name` to `--user-name` in `file-server create` command to be consistent with `cluster create` command

Billing

- Added enrollment account commands

Consumption

- Added `marketplace` commands
- [BREAKING CHANGE] Renamed `reservations summaries` to `reservation summary`
- [BREAKING CHANGE] Renamed `reservations details` to `reservation detail`
- [BREAKING CHANGE] Removed `--reservation-order-id` and `--reservation-id` short options for `reservation` commands
- [BREAKING CHANGE] Removed `--grain` short options for `reservation summary` commands
- [BREAKING CHANGE] Removed `--include-meter-details` short options for `pricesheet` commands

Container

- Added git repo volume mount parameters `--gitrepo-url` `--gitrepo-dir` `--gitrepo-revision` and `--gitrepo-mount-path`
- Fixed #5926: `az container exec` failing when `--container-name` specified

Extension

- Changed distribution check message to be debug-level

Interactive

- Changed to stop completions upon unrecognized commands
- Added event hooks before and after command subtree is created
- Added completion for `--ids` parameters

Network

- Fixed #5936: `application-gateway create` tags could not be set
- Added argument `--auth-certs` to attach authentication certificates for `application-gateway http-settings [create|update]`. #4910
- Added `ddos-protection` commands to create DDoS protection plans
- Added support for `--ddos-protection-plan` to `vnet [create|update]` to associate a VNet to a DDoS protection plan
- Fixed issue with `--disable-bgp-route-propagation` flag in `network route-table [create|update]`
- Removed dummy arguments `--public-ip-address-type` and `--subnet-type` for `network lb [create|update]`
- Added support for TXT records with RFC 1035 escape sequences to `network dns zone [import|export]` and `network dns record-set txt add-record`

Profile

- Added support for Azure Classic accounts in `account list`
- [BREAKING CHANGE] Removed `--msi` & `--msi-port` arguments

RDBMS

- Added `georestore` command
- Removed storage size restriction from `create` command

Resource

- Added support for `--metadata` to `policy definition create`
- Added support for `--metadata`, `--set`, `--add`, `--remove` to `policy definition update`

SQL

- Added `sql elastic-pool op list` and `sql elastic-pool op cancel`

Storage

- Improved error messages for malformed connection strings

VM

- Added support to configure platform fault domain count to `vmss create`
- Changed `vmss create` to default to Standard LB for zonal, large or single-placement-group disabled scale-set
- [BREAKING CHANGE]: Removed `vm assign-identity`, `vm remove-identity` and `vm format-secret``
- Added support for Public-IP SKU to `vm create`
- Added `--keyvault` and `--resource-group` arguments to `vm secret format` to support scenarios where the command is unable to resolve the vault ID. [#5718 ↗](#)
- Better errors for `[vm|vmss create]` when a resource group's location has no zone support

March 27, 2018

Version 2.0.30

Core

- Show message for extensions marked as preview in help

ACS

- Fix SSL certificate verification error for `aks install-cli` in Cloud Shell

Appservice

- Added HTTPS-only support to `webapp update`
- Added support for slots to `az webapp identity [assign|show]` and `az functionapp identity [assign|show]`

Backup

- Added new command `az backup protection isenabled-for-vm`. This command can be used to check if a VM is backed up by any vault in the subscription
- Enabled Azure object IDs for `--resource-group` and `--vault-name` parameters for the following commands:
 - `backup container show`
 - `backup item set-policy`
 - `backup item show`
 - `backup job show`
 - `backup job stop`
 - `backup job wait`
 - `backup policy delete`
 - `backup policy get-default-for-vm`
 - `backup policy list-associated-items`
 - `backup policy set`
 - `backup policy show`
 - `backup protection backup-now`
 - `backup protection disable`
 - `backup protection enable-for-vm`
 - `backup recoverypoint show`
 - `backup restore files mount-rp`
 - `backup restore files unmount-rp`
 - `backup restore restore-disks`
 - `backup vault delete`
 - `backup vault show`
- Changed `--name` parameters to accept the output format from `backup ... show` commands

Container

- Added `container exec` command. Executes commands in a container for a running container group
- Allow table output for creating and updating a container group

Extension

- Added message for `extension add` if extension is in preview
- Changed `extension list-available` to show full extension data with `--show-details`

- [BREAKING CHANGE] Changed `extension list-available` to show simplified extension data by default

Interactive

- Changed completions to activate as soon as command table loading is done
- Fixed bug with using `--style` parameter
- Interactive lexer instantiated after command table dump if missing
- Improved completer support

Lab

- Fixed bugs with `create environment` command

Monitor

- Added support for `--top`, `--orderby` and `--namespace` to `metrics list` [#5785 ↗](#)
- Fixed [#4529 ↗](#): `metrics list` Accepts a space-separated list of metrics to retrieve
- Added support for `--namespace` to `metrics list-definitions` [#5785 ↗](#)

Network

- Added support for Private DNS zones

Profile

- Added warning for `--identity-port` and `--msi-port` to `login`

RDBMS

- Added business model GA API version 2017-12-01

Resource

- [BREAKING CHANGE]: Changed `provider operation [list|show]` to not require `--api-version`

Role

- Added support for required access configurations and native clients to `az ad app create`
- Changed `rbac` commands to return less than 1000 IDs on object resolution
- Added credential management commands `ad sp credential [reset|list|delete]`
- [BREAKING CHANGE] Removed 'properties' from `az role assignment [list|show]` output
- Added support for `dataActions` and `notDataActions` permissions to `role definition`

Storage

- Fixed issue when uploading file with size between 195GB and 200GB
- Fixed [#4049](#): Problems with append blob uploads ignoring condition parameters

VM

- Added warning to `vmss create` for upcoming breaking changes for sets with 100+ instances
- Added zone resilient support to `vm [snapshot|image]`
- Changed disk instance view to report better encryption status
- [BREAKING CHANGE] Changed `vm extension delete` to no longer return output

March 13, 2018

Version 2.0.29

ACR

- Added support for `--image` parameter to `repository delete`
- Deprecated `--manifest` and `--tag` parameters of the `repository delete` command
- Added `repository untag` command to remove a tag without deleting data

ACS

- Added `aks upgrade-connector` command to upgrade an existing connector
- Changed `kubectl` config files to use a more readable block-style YAML

Advisor

- [BREAKING CHANGE] Renamed `advisor configuration get` to `advisor configuration list`
- [BREAKING CHANGE] Renamed `advisor configuration set` to `advisor configuration update`
- [BREAKING CHANGE] Removed `advisor recommendation generate`
- Added `--refresh` parameter to `advisor recommendation list`
- Added `advisor recommendation show` command

Appservice

- Deprecated `[webapp|functionapp] assign-identity`
- Added managed identity commands `webapp identity [assign|show]` and `functionapp identity [assign|show]`

Eventhubs

- Initial release

Extension

- Added check to warn user if used distro is different then the one stored in package source file, as this may lead into errors

Interactive

- Fixed [#5625](#): Persist history across different sessions
- Fixed [#3016](#): History not recorded while in scope
- Fixed [#5688](#): Completions did not appear if command table loading encountered an exception
- Fixed progress meter for long running operations

Monitor

- Deprecated the `monitor autoscale-settings` commands
- Added `monitor autoscale` commands
- Added `monitor autoscale profile` commands
- Added `monitor autoscale rule` commands

Network

- [BREAKING CHANGE] Removed `--tags` parameter from `route-filter rule create`
- Removed some erroneous default values for the following commands:
 - `network express-route update`
 - `network nsg rule update`
 - `network public-ip update`
 - `traffic-manager profile update`
 - `network vnet-gateway update`
- Added `network watcher connection-monitor` commands`
- Added `--vnet` and `--subnet` parameters to `network watcher show-topology`

Profile

- Deprecated `--msi` parameter for `az login`
- Added `--identity` parameter for `az login` to replace `--msi`

RDBMS

- [PREVIEW] Changed to use the API 2017-12-01-preview

Service Bus

- Initial release

Storage

- Fixed #4971: `storage blob copy` now supports other Azure clouds
- Fixed #5286: Batch commands `storage blob [delete-batch|download-batch|upload-batch]` no longer throw an error upon precondition failures

VM

- Added support to `[vm|vmss] create` to attach unmanaged data disks and configure caching
- Deprecated `[vm|vmss] assign-identity` and `[vm|vmss] remove-identity`
- Added `vm identity [assign|remove|show]` and `vmss identity [assign|remove|show]` commands to replace deprecated commands
- Changed default priority in `vmss create` to None

February 27, 2018

Version 2.0.28

Core

- Fixed [#5184](#): Homebrew install issue
- Added support for extension telemetry with custom keys
- Added HTTP logging to `--debug`

ACS

- Changed to use the the `virtual-kubelet-for-aks` Helm chart for `aks install-connector` by default
- Fixed issue: Insufficient permission for service principals to create ACI container group issue
- Added `--aci-container-group`, `--location`, and `--image-tag` parameters to `aks install-connector`
- Removed deprecation notice from `aks get-versions`

Appservice

- Updates for new SDK version (azure-mgmt-web 0.35.0)
- Fixed [#5538](#): `Free` reported as invalid SKU

Cognitive Services

- Updated the 'notice' when creating a new Cognitive Services account

Consumption

- Added new commands for pricesheet API
- Updated the existing Usage Details and Reservation Details formats

Container

- Added `--secrets` and `--secrets-mount-path` arguments to `container create` to use secrets in ACI

Network

- Fixed #5559: Missing client in `network vnet-gateway vpn-client generate`

Resource

- Changed `group deployment export` to display a partial template and errors on failure

Role

- Added `role assignment list-changelogs` to allow auditing of service principal roles

SQL

- Added zone redundancy support for databases and elastic pools on creation and update

Storage

- Enabled specifying destination-path/prefix for `storage blob [upload-batch|download-batch]`

VM

- Added support for attaching/detaching disks on a single VMSS instance

February 13, 2018

Version 2.0.27

Core

- Changed authentication to key on both subscription ID and name on MSI login

ACS

- [BREAKING CHANGE] Renamed `aks get-versions` to `aks get-upgrades` in the interest of accuracy
- Changed `aks get-versions` to show Kubernetes versions available for `aks create`
- Changed `aks create` defaults to letting the server choose the version of Kubernetes
- Updated help messages referring to the service principal generated by AKS
- Changed default node sizes for `aks create` from "Standard_D1_v2" to "Standard_DS1_v2"
- Improved reliability when locating the dashboard pod for `az aks browse`
- Fixed `aks get-credentials` to handle Unicode errors when loading Kubernetes configuration files
- Added a message to `az aks install-cli` to help get `kubectl` in `$PATH`

Appservice

- Fixed issue where `webapp [backup|restore]` failed because of a null reference
- Added support for default app service plans through `az configure --defaults appserviceplan=my-asp`

CDN

- Added `cdn custom-domain [enable-https|disable-https]` commands

Container

- Added `--follow` option to `az container logs` for streaming logs
- Added `container attach` command that attaches local standard output and error streams to a container in a container group

CosmosDB

- Added support for setting capabilities

Extension

- Added support for `--pip-proxy` parameter to `az extension [add|update]` commands
- Added support for `--pip-extra-index-urls` argument to `az extension [add|update]` commands

Feedback reference

- Added extension information to telemetry data

Interactive

- Fixed issue where user is prompted to login when using interactive mode in Cloud Shell
- Fixed regression with missing parameter completions

IoT

- Fixed issue where `iot dps access policy [create|update]` would return a 'not found' error on success
- Fixed issue where `iot dps linked-hub [create|update]` would return a 'not found' error on success
- Added `--no-wait` support to `iot dps access policy [create|update]` and `iot dps linked-hub [create|update]`
- Changed `iot hub create` to allow specifying the number of partitions

Monitor

- Fixed `az monitor log-profiles create` command

Network

- Fixed the `--tags` option for the following commands:
 - `network public-ip create`
 - `network lb create`
 - `network local-gateway create`
 - `network nic create`
 - `network vnet-gateway create`
 - `network vpn-connection create`

Profile

- Enabled `az login` in from interactive mode

Resource

- Added back `feature show`

Role

- Added `--available-to-other-tenants` argument to `ad app update`

SQL

- Added `sql server dns-alias` commands
- Added `sql db rename`
- Added support for the `--ids` argument to all sql commands

Storage

- Added `storage blob service-properties delete-policy` and `storage blob undelete` commands to enable soft-delete

VM

- Fixed a crash when VM encryption may not be fully initialized
- Added principal ID output on enabling MSI
- Fixed `vm boot-diagnostics get-boot-log`

January 31, 2018

Version 2.0.26

Core

- Added support raw token retrieval in MSI context
- Removed polling indicator string after finishing LRO on Windows cmd.exe
- Added a warning that appears when using a configured default has been changed to an INFO level entry. Use `--verbose` to see
- Add a progress indicator for wait commands

ACS

- Clarified `--disable-browser` argument
- Improved tab completion for `--vm-size` arguments

Appservice

- Fixed `webapp log [tail|download]`
- Removed the `kind` check on webapps and functions

CDN

- Fixed missing client issue with `cdn custom-domain create`

CosmosDB

- Fixed parameter description for failover policies

Interactive

- Fixed issue where command option completions no longer appeared

Network

- Added protection for `--cert-password` to `application-gateway create`
- Fixed issue with `application-gateway update` where `--sku` erroneously applied a default value
- Added protection for `--shared-key` and `--authorization-key` to `vpn-connection create`
- Fixed missing client issue with `asg create`
- Added `--file-name / -f` parameter for exported names to `dns zone export`
- Fixed the following issues with `dns zone export`:
 - Fixed issue where long TXT records were incorrectly exported
 - Fixed issue where quoted TXT records were incorrectly exported without escaped quotes
- Fixed issue where certain records were imported twice with `dns zone import`
- Restored `vnet-gateway root-cert` and `vnet-gateway revoked-cert` commands

Profile

- Fixed `get-access-token` to work inside a VM with identity

Resource

- Fixed bug with `deployment [create|validate]` where warning was incorrectly displayed when a template 'type' field contained uppercase values

Storage

- Fixed issue with migrating Storage V1 accounts to Storage V2
- Added progress reporting for all upload/download commands
- Fixed bug preventing "-n" arg option with `storage account check-name`
- Added 'snapshot' column to table output for `blob [list|show]`
- Fixed bugs with various parameters that needed to be parsed as ints

VM

- Added `vm image accept-terms` command to allow creating VMs from images with additional charges
- Fixed `[vm|vmss create]` to ensure commands can run under proxy with unsigned certificates
- [PREVIEW] Added support for "low" priority to VMSS
- Added protection for `--admin-password` to `[vm|vmss] create`

January 17, 2018

Version 2.0.25

ACR

- Added acr login fallback on Windows credential errors
- Enabled registry logs

ACS

- Fixed `get-credentials` command
- Removed SPN role requirement

Appservice

- Fixed bug with `config ssl upload` where `hosting_environment_profile` was null
- Added support for custom URLs to `browse`
- Fixed slot support for `log tail`

Backup

- Changed `--container-name` option of `backup item list` to be optional
- Added storage account options to `backup restore restore-disks`
- Fixed location check in `backup protection enable-for-vm` to be case insensitive
- Fixed issue where commands failed with an invalid container name
- Changed `backup item list` to include 'Health Status' by default

Batch

- Changed `batch login` to return authentication details

Cloud

- Changed to not require endpoints when setting `--profile` on a cloud

Consumption

- Added new commands for reservations: `consumption reservations summaries` and `consumption reservations details`

Event Grid

- [BREAKING CHANGE] Moved the `az eventgrid topic event-subscription` commands to `eventgrid event-subscription`
- [BREAKING CHANGE] Moved the `az eventgrid resource event-subscription` commands to `eventgrid event-subscription`
- [BREAKING CHANGE] Removed the `eventgrid event-subscription show-endpoint-url` command. Use `eventgrid event-subscription show --include-full-endpoint-url` instead
- Added command `eventgrid topic update`
- Added command `eventgrid event-subscription update`
- Added `--ids` parameter for `eventgrid topic` commands
- Added tab completion support for topic names

Interactive

- Fixed issue where interactive mode did not work with Python 2.x
- Fixed errors on startup

- Fixed issue with some commands not running in interactive mode

IoT

- Added support for device provisioning service
- Added deprecation messages in commands and command help
- Added IoT check to inform users of the IoT Extension

Monitor

- Added multi-diagnostic setting support. The `--name` parameter is now required for
`az monitor diagnostic-settings create`
- Added command `monitor diagnostic-settings categories` to get diagnostic settings category

Network

- Fixed issue when trying to change to/from active-standby mode with `vnet-gateway update`
- Added support for HTTP2 to `application-gateway [create|update]`

Profile

- Added support for login with user assigned identities

Role

- Added `--assignee-object-id` argument to `role assignment create` to bypass graph query

Service Fabric

- Added detailed errors to validation response when creating cluster
- Fixed missing client issue with several commands

VM

- [PREVIEW] Cross-zone support for `vmss`

- [BREAKING CHANGE] Changed single-zone `vmss` default to "Standard" load balancer
- [BREAKING CHANGE] Changed `externalIdentities` to `userAssignedIdentities` for EMSI
- [PREVIEW] Added support for OS disk swap
- Added support for using VM images from other subscriptions
- Added `--plan-name`, `--plan-product`, `--plan-promotion-code` and `--plan-publisher` arguments to `[vm|vmss] create`
- Fixed error issues with `[vm|vmss] create`
- Fixed excessive resource usage caused by `vm image list --all`

December 19, 2017

Version 2.0.23

- Added support for login with user assigned identities

Container

- Fixed incorrect order of parameters for container logs

Network

- Added `--disable-bgp-route-propagation` argument to `route-table [create|update]`
- Added `--ip-tags` argument to `public-ip [create|update]`

Storage

- Added support for storage V2

VM

- [PREVIEW] Added support for user-assigned identities for VMs and VMSSes

December 5, 2017

Version 2.0.22

- Removed `az component` commands. Use `az extension` instead

Core

- Modified the `AZURE_US_GOV_CLOUD` AAD authority endpoint from `login.microsoftonline.com` to `login.microsoftonline.us`
- Fixed issue where telemetry would continuously resend

ACS

- Added `aks install-connector` and `aks remove-connector` commands
- Improved error reporting for `acs create`
- Fixed usage of `aks get-credentials -f` without fully-qualified path

Advisor

- Initial release

Appservice

- Fixed cert name generation with `webapp config ssl upload`
- Fixed `webapp [list|show]` and `functionapp [list|show]` to display correct apps
- Added default value for `WEBSITE_NODE_DEFAULT_VERSION`

Consumption

- Added support for API version 2017-11-30

Container

- Fixed default ports regression

Monitor

- Added multi-dimension support to metrics command

Resource

- Added `--include-response-body` argument to `resource show`

Role

- Added display of default assignments for "classic" administrators to `role assignment list`
- Added support to `ad sp reset-credentials` for adding credentials instead of overwriting
- Improved error reporting for `ad sp create-for-rbac`

SQL

- Added `sql db list-usages` and `sql db show-usage` commands
- Added `sql server conn-policy show` and `sql server conn-policy update` commands

VM

- Added zone information to `az vm list-skus`

November 14, 2017

Version 2.0.21

ACR

- Added support for creating webhooks in replication regions

ACS

- Changed all wording of "agent" to "node" in AKS
- Deprecated `--orchestrator-release` option for `acs create`
- Changed default VM size for AKS to `standard_D1_v2`
- Fixed `az aks browse` on Windows
- Fixed `az aks get-credentials` on Windows

Appservice

- Added deployment source `config-zip` for webapps and function apps
- Added `--docker-container-logging` option to `az webapp log config`

- Removed the `storage` option from the parameter `--web-server-logging` of `az webapp log config`
- Improved error messages for `deployment user set`
- Added support for creating Linux function apps
- Fixed `list-locations`

Batch

- Fixed bug in pool create command when a resource ID was used with the `--image` flag

Batchai

- Added short option, `-s`, for `--vm-size` when providing VM size in `file-server create` command
- Added storage account name and key arguments to `cluster create` parameters
- Fixed documentation for `job list-files` and `job stream-file`
- Added short option, `-r`, for `--cluster-name` when providing cluster name in `job create` command

Cloud

- Changed `cloud [register|update]` to prevent registering clouds that have missing required endpoints

Container

- Added support to open multiple ports
- Added container group restart policy
- Added support to mount Azure File share as a volume
- Updated helper docs

Data Lake Analytics

- Changed `[job|account] list` to return more concise information

Data Lake Store

- Changed `account list` to return more concise information

Extension

- Added `extension list-available` to allow listing official Microsoft extensions
- Added `--name` to `extension [add|update]` to allow installing extensions by name

IoT

- Added support for certificate authorities (CA) and certificate chains

Monitor

- Added `activity-log alert` commands

Network

- Added support for CAA DNS records
- Fixed issue where endpoints could not be updated with `traffic-manager profile update`
- Fixed issue where `vnet update --dns-servers` didn't work depending on how the VNET was created
- Fixed issue where relative DNS names were incorrectly imported by `dns zone import`

Reservations

- Initial preview release

Resource

- Added support for resource IDs to `--resource` parameter and resource-level locks

SQL

- Added `--ignore-missing-vnet-service-endpoint` parameter to `sql server vnet-rule [create|update]`

Storage

- Changed `storage account create` to use SKU `Standard_RAGRS` as default

- Fixed bugs when dealing with file/blob names that include non-ascii chars
- Fixed bug that prevented using `--source-uri` with `storage [blob|file] copy start-batch`
- Added commands to glob and delete multiple objects with `storage [blob|file] delete-batch`
- Fixed issue when enabling metrics with `storage metrics update`
- Fixed issue with files over 200GB when using `storage blob upload-batch`
- Fixed issue where `--bypass` and `--default-action` were ignored by `storage account [create|update]`

VM

- Fixed a bug with `vmss create` that prevented using the `Basic` size tier
- Added `--plan` arguments to `[vm|vmss] create` for custom images with billing information
- Added `vm secret [add|remove|list]` commands`
- Renamed `vm format-secret` to `vm secret format`
- Added `--encrypt format` argument to `vm encryption enable`

October 24, 2017

Version 2.0.20

Core

- Updated `2017-03-09-profile` to consume `MGMT_STORAGE` API version `2016-01-01`

ACR

- Updated resource management to point to `2017-10-01` API version
- Changed 'bring your own storage' SKU to Classic
- Renamed registry SKUs to Basic, Standard, and Premium

ACS

- [PREVIEW] Added `az aks` commands
- Fixed kubernetes `get-credentials`

Appservice

- Fixed issue where downloaded `webapp` logs may be invalid

Component

- Added clearer deprecation message for all installers and confirmation prompt

Monitor

- Added `action-group` commands

Resource

- Fixed incompatibility with most recent version of msrest dependency in `group export`
- Fixed `policy assignment create` to work with built in policy definitions and policy set definitions

VM

- Added `--accelerated-networking` argument to `vmss create`

October 9, 2017

Version 2.0.19

Core

- Added handling of ADFS authority URLs with a trailing slash to Azure Stack

Appservice

- Added generic update with new command `webapp update`

Batch

- Updated to Batch SDK 4.0.0

- Updated `--image` option of `VirtualMachineConfiguration` to support ARM image references in addition to `publish:offer:sku:version`
- Added support for the new CLI extension model for Batch Extensions commands
- Removed Batch support from the component model

Batchai

- Initial release of Batch AI module

Keyvault

- Fixed Key Vault authentication issue when using ADFS on Azure Stack. ([#4448](#)) ↗

Network

- Changed `--server` argument of `application-gateway address-pool create` to be optional, allowing for empty address pools
- Updated `traffic-manager` to support latest features

Resource

- Added support for `--resource-group/-g` options for resource group name to `group`
- Added commands for `account lock` to work with subscription-level locks
- Added commands for `group lock` to work with group-level locks
- Added commands for `resource lock` to work with resource-level locks

Sql

- Added support for SQL Transparent Data Encryption (TDE) and TDE with Bring Your Own Key
- Added `db list-deleted` command and `db restore --deleted-time` parameter, allowing the ability to find and restore deleted databases
- Added `db op list` and `db op cancel`, allowing the ability to list and cancel in-progress operations on database

Storage

- Added support for file share snapshot

Vm

- Fixed a bug in `vm show` where using `-d` caused a crash on missing private ip addresses
- [PREVIEW] Added support for rolling upgrade to `vmss create`
- Added support for updating encryption settings with `vm encryption enable`
- Added `--os-disk-size-gb` parameter to `vm create`
- Added `--license-type` parameter for Windows to `vmss create`

September 22, 2017

Version 2.0.18

Resource

- Added support for showing built-in policy definitions
- Added support mode parameter for creating policy definitions
- Added support for UI definitions and templates to `managedapp definition create`
- [BREAKING CHANGE] Changed `managedapp` resource type from `appliances` to `applications` and `applianceDefinitions` to `applicationDefinitions`

Network

- Added support for availability zone to `network 1b` and `network public-ip` subcommands
- Added support for IPv6 Microsoft Peering to `express-route`
- Added `asg` application security group commands
- Added `--application-security-groups` argument to `nic [create|ip-config create|ip-config update]`
- Added `--source-asgs` and `--destination-asgs` arguments to `nsg rule [create|update]`
- Added `--ddos-protection` and `--vm-protection` arguments to `vnet [create|update]`
- Added `network [vnet-gateway|vpn-client|show-url]` commands

Storage

- Fixed issue where `storage account network-rule` commands may fail after updating the SDK

Eventgrid

- Updated Azure Event Grid Python SDK to use newer API version "2017-09-15-preview"

SQL

- Changed `sql server list` argument `--resource-group` to be optional. If not specified, all sql servers in the subscription will be returned
- Added `--no-wait` param to `db [create|copy|restore|update|replica create|create|update]` and `dw [create|update]`

Keyvault

- Added support for Keyvault commands from behind a proxy

VM

- Added for support to availability zone to `[vm|vmss|disk] create`
- Fixed issue where using `--app-gateway ID` with `vmss create` would cause a failure
- Added `--asgs` argument to `vm create`
- Added support for running commands on VMs with `vm run-command`
- [PREVIEW] Added support for VMSS disk encryption with `vmss encryption`
- Added support for performing maintenance on VMs with `vm perform-maintenance`

ACS

- [PREVIEW] Added `--orchestrator-release` argument to `acs create` for ACS preview regions

Appservice

- Added ability to update and show authentication settings with `webapp auth [update|show]`

Backup

- Preview release

September 11, 2017

Version 2.0.17

Core

- Enabled command module to set its own correlation ID in telemetry
- Fixed JSON dump issue when telemetry is set to diagnostics mode

Acs

- Added `acs list-locations` command
- Made `ssh-key-file` come with expected default value

Appservice

- Added ability to create a webapp in a resource group other than the active service plan's

CDN

- Fixed 'CustomDomain is not interable' bug for `cdn custom-domain create`

Extension

- Initial Release

Keyvault

- Fixed issue where permissions were case sensitive for `keyvault set-policy`

Network

- Renamed `vnet list-private-access-services` to `vnet list-endpoint-services`
- Renamed `--private-access-services` argument to `--service-endpoints` for `vnet subnet create/update`
- Added support for multiple IP ranges and port ranges to `nsg rule create/update`
- Added support for SKU to `lb create`
- Added support for SKU to `public-ip create`

Resource

- Allow passing in resource policy parameter definitions in `policy definition create`, and `policy definition update`
- Allow passing in parameter values for `policy assignment create`
- Allow for passing JSON or file for all params
- Incremented API version

SQL

- Added `sql server vnet-rule` commands

VM

- Fixed: Don't assign access unless `--scope` is provided
- Fixed: Use the same extension naming as portal does
- Removed `subscription` from the `[vm|vmss] create` output
- Fixed: `[vm|vmss] create` storage SKU is not applied on data disks with an image
- Fixed: `vm format-secret --secrets` would not accept newline separated IDs

August 31, 2017

Version 2.0.16

Keyvault

- Fixed bug when trying to automatically resolve secret encoding with `secret download`

Sf

- Deprecating all commands in favor of Service Fabric CLI (sfctl)

Storage

- Fixed issue where storage accounts could not be created in regions that don't support the NetworkACLs feature
- Determine content type and content encoding during blob and file upload if neither content type and content encoding are specified

August 28, 2017

Version 2.0.15

CLI

- Added legal note to `--version`

ACS

- Corrected preview regions
- Formatted default `dns_name_prefix` properly
- Optimized acs command output

Appservice

- [BREAKING CHANGE] Fixed inconsistencies in the output of `az webapp config appsettings [delete|set]`
- Added a new alias of `-i` for `az webapp config container set --docker-custom-image-name`
- Exposed `az webapp log show`
- Exposed new arguments from `az webapp delete` to retain app service plan, metrics or dns registration
- Fixed: Detect slot settings correctly

IoT

- Fixed #3934: Policy creation no longer clears existing policies

Network

- [BREAKING CHANGE] Renamed `vnet list-private-access-services` to `vnet list-endpoint-services`
- [BREAKING CHANGE] Renamed option `--private-access-services` to `--service-endpoints` for `vnet subnet [create|update]`
- Added support for multiple IP and port ranges to `nsg rule [create|update]`
- Added support for SKU to `lb create`
- Added support for SKU to `public-ip create`

Profile

- Exposed `--msi` and `--msi-port` to login using a virtual machine's identity

Service Fabric

- Preview release
- Simplified registry user/password rules for command
- Fixed password prompt for user even after passing in the param
- Added support for empty `registry_cred`

Storage

- Enabled setting blob tier
- Added `--bypass` and `--default-action` arguments to `storage account [create|update]` to support service tunneling
- Added commands to add VNET rules and IP based rules to `storage account network-rule`
- Enabled service encryption by customer managed key
- [BREAKING CHANGE] Renamed `--encryption` option to `--encryption-services` for `az storage account create` and `az storage account update` command
- Fixed #4220: `az storage account update encryption` - syntax mismatch

VM

- Fixed issue where extra, erroneous information was displayed for `vmss get-instance-view` when using `--instance-id *`
- Added support for `--lb-sku` to `vmss create`
- Removed human names from the admin names disallowed for `[vm|vmss] create`
- Fixed issue where `[vm|vmss] create` would throw an error if unable to extract plan information from an image
- Fixed a crash when creating a vmms scaleset with an internal LB
- Fixed issue where `--no-wait` argument did not work wth `vm availability-set create`

August 15, 2017

Version 2.0.14

ACS

- Corrected sshMaster0 port number for kubernetes

Appservice

- Fixed an exception when creating a new git based Linux webapp

Event Grid

- Added SDK dependencies

August 11, 2017

Version 2.0.13

ACS

- Added more preview regions

Batch

- Updated to Batch SDK 3.1.0 and Batch Management SDK 4.1.0
- Added a new command show the task counts of a job
- Fixed bug in resource file SAS URL processing
- Batch account endpoint now supports optional 'https://' prefix
- Support for adding lists of more than 100 tasks to a job
- Added debug logging for loading Extensions command module

Component

- Added deprecation warning to 'az component' commands

Container

- `create`: Fixed issue where equals sign was not allowed inside an environment variable

Data Lake Store

- Enabled progress control

Event Grid

- Initial release

Network

- `lb`: Fixed issue where the certain child resource names did not resolve correctly when omitted
- `application-gateway {subresource} delete`: Fixed issue where `--no-wait` was not honored
- `application-gateway http-settings update`: Fixed issue where `--connection-draining-timeout` could not be turned off
- Fixed error unexpected keyword argument `sa_data_size_kilobytes` with `az network vpn-connection ipsec-policy add`

Profile

- `account list`: Added `--refresh` to sync up the latest subscriptions from server

Storage

- Enable update storage account with system assigned identity

VM

- `availability-set`: Exposed fault domain count on convert
- Exposed `list-skus` command
- Support to assign identity w/o creating role assignments
- Apply storage sku on attaching data disks
- Removed default os-disk name and storage SKU when using managed disks

July 28, 2017

Version 2.0.12

- Added container commands
- Added billing and consumption modules

text

```
azure-cli (2.0.12)

acr (2.0.9)
acs (2.0.11)
appservice (0.1.11)
batch (3.0.3)
billing (0.1.3)
cdn (0.0.6)
cloud (2.0.7)
cognitiveservices (0.1.6)
command-modules-nspkg (2.0.1)
component (2.0.6)
configure (2.0.10)
consumption (0.1.3)
container (0.1.7)
core (2.0.12)
cosmosdb (0.1.11)
dla (0.0.10)
dls (0.0.11)
feedback (2.0.6)
find (0.2.6)
interactive (0.3.7)
iot (0.1.10)
keyvault (2.0.8)
lab (0.0.9)
monitor (0.0.8)
network (2.0.11)
nspkg (3.0.1)
profile (2.0.9)
rdbms (0.0.5)
redis (0.2.7)
resource (2.0.11)
role (2.0.9)
sf (1.0.5)
sql (2.0.8)
storage (2.0.11)
vm (2.0.11)
```

Core

- Output sdk auth info for service principals with certificates
- Fixed deployment progress exceptions
- Use arm endpoint from the current cloud to create subscription client
- Improved concurrent handling of clouds.config file (#3636)
- Refresh client request id for each command execution
- Create subscription clients with right SDK profile (#3635)
- Progress Reporting for template deployments (#3510)
- Added support for picking table output fields through jmespath query (#3581)

- Improved the muting of parse args and append history with gestures (#3434)
- Create subscription clients with right SDK profile
- Move all existing recording files to latest folder
- Fixed idempotency for VM/VMSS create (#3586)
- Command paths are no longer case sensitive
- Certain boolean-type parameters are no longer case sensitive
- Support login to ADFS on prem server like Azure Stack
- Fixed concurrent writes to clouds.config (#3255)

ACR

- Added `show-usage` command for managed registries
- Support SKU update for managed registries
- Added managed registries with managed SKU
- Added webhooks for managed registries with acr webhook command module
- Added AAD authentication with acr login command
- Added delete command for docker repositories, manifests, and tags

ACS

- Support for API version 2017-07-01

Appservice

- Fixed bug where listing Linux webapp would return nothing
- Support to retrieve creds from acr
- Remove all commands under `appservice web`
- Mask docker registry passwords from command output (#3656)
- Ensure default browser is used on macOS without errors (#3623)
- Improve the help of `webapp log tail` and `webapp log download` (#3624)
- Exposed `traffic-routing` command to configure static routing (#3566)
- Added reliability fixes in configuring source control (#3245)
- Removed unsupported `--node-version` argument from `webapp config update` for Windows webapps. Instead use `webapp config appsettings set --settings WEBSITE_NODE_DEFAULT_VERSION=...`

Batch

- Updated to Batch SDK 3.0.0 with support for low-priority VMs in pools
- Renamed `pool create` option `--target-dedicated` to `--target-dedicated-nodes`

- Added `pool create` options `--target-low-priority-nodes` and `--application-licenses`

CDN

- Provided a better error message for `cdn endpoint list` when the profile specified by `--profile-name` does not exist

Cloud

- Changed API version of cloud metadata endpoint to YYYY-MM-DD format
- Gallery endpoint isn't required
- Support for registering cloud just with ARM resource manager endpoint
- Provided an option for `cloud set` to choose the profile while selecting current cloud
- Exposed `endpoint_vm_image_alias_doc`

CosmosDB

- Fixed allowing creation of collection with custom partition key
- Added support for collection default TTL

Data Lake Analytics

- Added commands for compute policy management under the `dla account compute-policy` heading
- Added `dla job pipeline show`
- Added `dla job recurrence list`

Data Lake Store

- Added support for user managed key vault key rotation in `dls account update`
- Updated underlying Data Lake Store filesystem SDK version, addressing a performance issue
- Added command `dls enable-key-vault`. This command attempts to enable a user provided Key Vault for use encrypting the data in a Data Lake Store account

Interactive

- Improved the start up time by using cached commands
- Increased test coverage
- Enhanced the '?' gesture to also inject into the next command
- Fixed interactive errors with the profile 2017-03-09-profile-preview (#3587)
- Allowed `--version` as a parameter for interactive mode (#3645)
- Stop interactive mode throwing errors from validation completions (#3570)
- Progress reporting for template deployments (#3510)
- Added `--progress` flag
- Removed `--debug` and `--verbose` from completions
- Removed `interactive` from completions (#3324)

IoT

- Fixed policy creation no longer clears existing policies. (#3934)

Key vault

- Added commands for key vault recovery features:
 - `keyvault subcommands purge, recover, keyvault list-deleted`
 - `keyvault secret subcommands backup, restore, purge, recover, list-deleted`
 - `keyvault certificate subcommands purge, recover, list-deleted`
 - `keyvault key subcommands purge, recover, list-deleted`
- Added service principal key vault integration (#3133)
- Updated key vault dataplane to 0.3.2. (#3307)

Lab

- Added support for claiming any vm in the lab through `az lab vm claim`
- Added table output formatter for `az lab vm list` and `az lab vm show`

Monitor

- Fix for template file with `monitor autoscale-settings get-parameters-template` command (#3349)
- Renamed `monitor alert-rule-incidents list` to `monitor alert list-incidents`
- Renamed `monitor alert-rule-incidents show` to `monitor alert show-incident`
- Renamed `monitor metric-defintions list` to `monitor metrics list-definitions`
- Renamed `monitor alert-rules` to `monitor alert`
- Changed `monitor alert create`:

- `condition` and `action` subcommands no longer accept JSON
- Add numerous parameters to simplify the rule creation process
- `location` no longer required
- Add name and ID support for target
- Remove `--alert-rule-resource-name`
- Rename `is-enabled` to `enabled`, no longer required
- `description` defaults now based on the supplied condition
- Add examples to help clarify the new format
- Support names or IDs for `monitor metric` commands
- Added convenience arguments and examples to `monitor alert rule update`

Network

- Added `list-private-access-services` command
- Added `--private-access-services` argument to `vnet subnet create` and `vnet subnet update`
- Fixed issue where `application-gateway redirect-config create` would fail
- Fixed issue where `application-gateway redirect-config update` with `--no-wait` would not work
- Fixed bug when using `--servers` argument with `application-gateway address-pool create` and `application-gateway address-pool update`
- Added `application-gateway redirect-config` commands
- Added commands to `application-gateway ssl-policy`: `list-options`, `predefined list`, `predefined show`
- Added arguments to `application-gateway ssl-policy set`: `--name`, `--cipher-suites`, `--min-protocol-version`
- Added arguments to `application-gateway http-settings create` and `application-gateway http-settings update`: `--host-name-from-backend-pool`, `--affinity-cookie-name`, `--enable-probe`, `--path`
- Added arguments to `application-gateway url-path-map create` and `application-gateway url-path-map update`: `--default-redirect-config`, `--redirect-config`
- Added argument `--redirect-config` to `application-gateway url-path-map rule create`
- Added support for `--no-wait` to `application-gateway url-path-map rule delete`
- Added arguments to `application-gateway probe create` and `application-gateway probe update`: `--host-name-from-http-settings`, `--min-servers`, `--match-body`, `--match-status-codes`

- Added argument `--redirect-config` to `application-gateway rule create` and `application-gateway rule update`
- Added support for `--accelerated-networking` to `nic create` and `nic update`
- Removed `--internal-dns-name-suffix` argument from `nic create`
- Added support for `--dns-servers` to `nic update` and `nic create`: Add support for `--dns-servers`
- Fixed bug where `local-gateway create` ignored `--local-address-prefixes`
- Added support for `--dns-servers` to `vnet update`
- Fixed bug when creating a peering without route filtering with `express-route peering create`
- Fixed bug where `--provider` and `--bandwidth` arguments did not work with `express-route update`
- Fixed bug with `network watcher show-topology` defaulting logic
- Improved output formatting for `network list-usages`
- Use default frontend IP for `application-gateway http-listener create` if only one exists
- Use default address pool, HTTP settings, and HTTP listener for `application-gateway rule create` if only one exists
- Use default frontend IP and backend pool for `lb rule create` if only one exists
- Use default frontend IP for `lb inbound-nat-rule create` if only one exists

Profile

- Support login inside a VM with a managed identity
- Support output for `account show` in SDK auth file format
- Show deprecation warnings when using '--expanded-view'
- Added `get-access-token` command to provide raw AAD token
- Support login with a user account with no associated subscriptions

RDBMS

- Support listing servers across a subscription (#3417)
- Fixed `%s` not processed because of missing `% server_type` (#3393)
- Fixed doc source map and added CI task to verify (#3361)
- Fixed MySQL and PostgreSQL help (#3369)

Resource

- Improved prompts for missing parameters for `group deployment create`

- Improved parsing of `--parameters KEY=VALUE` syntax
- Fixed issues where `group deployment create` parameter files were no longer recognized using `@<file>` syntax
- Support `--ids` argument for `resource` and `managedapp` commands
- Fixed up some parsing and error messages (#3584)
- Fixed `--resource-type` parsing for the `lock` command to accept `<resource-namespace>` and `<resource-type>`
- Added parameter checking for template link templates (#3629)
- Added support for specifying deployment parameters using `KEY=VALUE` syntax

Role

- Support output in SDK auth file format for `create-for-rbac`
- Cleaned up role assignments and related AAD application when deleting a service principal (#3610)
- Include time format in `app create` args `--start-date` and `--end-date` descriptions
- Show deprecation warnings when using `--expanded-view`
- Added key vault integration to the `create-for-rbac` and `reset-credentials` commands

Service Fabric

- Fixed an issue with large files in applications being truncated on upload (#3666)
- Added tests for Service Fabric commands (#3424)
- Fixed numerous Service Fabric commands (#3234)

SQL

- Removed broken `sql server create --identity` parameter
- Removed password values from `sql server create` and `sql server update` command output
- Added commands `sql db list-editions` and `sql elastic-pool list-editions`

Storage

- Removed `--marker` option from `storage blob list`, `storage container list`, and `storage share list` commands (#3745)
- Enabled creating an https-only storage account
- Updated storage metrics, logging and cors commands (#3495)

- Rephrased exception message from CORS add (#3638) (#3362)
- Converted generator to a list in download batch command dry run mode (#3592)
- Fixed blob download batch dryrun issue (#3640) (#3592)

VM

- Support configuring nsg
- Fixed a bug where the DNS server would not be configured correctly
- Support managed service identities
- Fixed issue where `cmss create` with an existing load balancer required `--backend-pool-name`
- Make datadisks created with `vm image create lun start with 0`

May 10, 2017

Version 2.0.6

- documentdb renamed to cosmosdb
- Add rdbms (mysql, postgres)
- Include Data Lake Analytics and Data Lake Store modules
- Include Cognitive Services module
- Include Service Fabric module
- Include Interactive module (rename of az-shell)
- Add support for CDN commands
- Remove Container module
- Add 'az -v' as shortcut for 'az --version' ([#2926](#))
- Improve performance of package load and command execution ([#2819](#))

```
text

azure-cli (2.0.6)

acr (2.0.4)
acs (2.0.6)
appservice (0.1.6)
batch (2.0.4)
cdn (0.0.2)
cloud (2.0.2)
cognitiveservices (0.1.2)
command-modules-nspkg (2.0.0)
component (2.0.4)
configure (2.0.6)
core (2.0.6)
cosmosdb (0.1.6)
dla (0.0.6)
```

```
dls (0.0.6)
feedback (2.0.2)
find (0.2.2)
interactive (0.3.1)
iot (0.1.5)
keyvault (2.0.4)
lab (0.0.4)
monitor (0.0.4)
network (2.0.6)
nspkg (3.0.0)
profile (2.0.4)
rdbms (0.0.1)
redis (0.2.3)
resource (2.0.6)
role (2.0.4)
sf (1.0.1)
sql (2.0.3)
storage (2.0.6)
vm (2.0.6)
```

Core

- core: capture exceptions caused by unregistered provider and auto-register it
- perf: persist adal token cache in memory till process exits ([#2603 ↗](#))
- Fix bytes returned from hex fingerprint -o tsv ([#3053 ↗](#))
- Enhanced Key Vault Certificate Download and AAD SP Integration ([#3003 ↗](#))
- Add Python location to 'az —version' ([#2986 ↗](#))
- login: support login when there are no subscriptions ([#2929 ↗](#))
- core: fix a failure when login using a service principal twice ([#2800 ↗](#))
- core: Allow file path of accessTokens.json to be configurable through an env var ([#2605 ↗](#))
- core: Allow configured defaults to apply on optional args ([#2703 ↗](#))
- core: Improved performance
- core: Custom CA Certs - Support setting REQUESTS_CA_BUNDLE environment variable
- core: Cloud configuration - use 'resource manager' endpoint if 'management' endpoint not set

ACS

- fix the master and agent count to be integer instead of string
- expose 'az acs create --no-wait' and 'az acs wait' for async creation
- expose 'az acs create --validate' for dry-run validations
- remove windows profile before PUT call for scale command ([#2755 ↗](#))

AppService

- functionapp: add full functionapp supports, including create, show, list, delete, hostname, ssl, etc
- Adding Team Services (vsts) as a continuous delivery option to "appservice web source-control config"
- Create "az webapp" to replace "az appservice web" (for backward compat, "az appservice web" will stay for 2 releases)
- Expose arguments to configure deployment and "runtime stacks" on webapp create
- Expose "webapp list-runtimes"
- support configure connection strings ([#2647 ↗](#))
- support slot swap with preview
- Polish errors from appservice commands ([#2948 ↗](#))
- Use the app service plan's resource group for cert operations ([#2750 ↗](#))

CosmosDB

- Rename documentdb module to cosmosdb
- Added support for documentdb data-plane APIs: database and collection management
- Added support for enabling automatic failover on database accounts
- Added support for new consistency policy ConsistentPrefix

Data Lake Analytics

- Fix a bug where filtering on result and state for job lists would throw an error
- Add support for new catalog item type: package. accessed through: `az dla catalog package`
- Made it possible to list the following catalog items from within a database (no schema specification required):
 - Table
 - Table valued function
 - View
 - Table Statistics. This can also be listed with a schema, but without specifying a table name

Data Lake Store

- Update the version of the underlying filesystem SDK, which gives better support for handling server side throttling scenarios
- Improve performance of package load and command execution ([#2819 ↴](#))
- missed help for access show. adding it. ([#2743 ↴](#))

Find

- improve search results and allow for versioning of the search index

KeyVault

- BC: `az keyvault certificate download` change -e from string or binary to PEM or DER to better represent the options
- BC: Remove --expires and --not-before from `keyvault certificate create` as these parameters are not supported by the service
- Adds the --validity parameter to `keyvault certificate create` to selectively override the value in --policy
- Fixes issue in `keyvault certificate get-default-policy` where 'expires' and 'not_before' were exposed but 'validity_in_months' was not
- keyvault fix for import of pem and pfx ([#2754 ↴](#))

Lab

- Adding create, show, delete & list commands for environment in the lab
- Adding show & list commands to view ARM templates in the lab
- Adding --environment flag in `az lab vm list` to filter VMs by environment in the lab
- Add convenience command `az lab formula export-artifacts` to export artifact scaffold within a Lab's formula
- Add commands to manage secrets within a Lab

Monitor

- Bug Fix: Modeling `--actions` of `az alert-rules create` to consume JSON string ([#3009 ↴](#))
- Bug fix - diagnostic settings create does not accept logs/metrics from show commands ([#2913 ↴](#))

Network

- Add `network watcher test-connectivity` command
- Add support for `--filters` parameter for `network watcher packet-capture create`
- Add support for Application Gateway connection draining
- Add support for Application Gateway WAF rule set configuration
- Add support for ExpressRoute route filters and rules
- Add support for TrafficManager geographic routing
- Add support for VPN connection policy-based traffic selectors
- Add support for VPN connection IPSec policies
- Fix bug with `vpn-connection create` when using the `--no-wait` or `--validate` parameters
- Add support for active-active VNet gateways
- Remove nulls values from output of `network vpn-connection list/show` commands
- BC: Fix bug in the output of `vpn-connection create`
- Fix bug where '--key-length' argument of 'vpn-connection create' was not parsed correctly
- Fix bug in `dns zone import` where records were not imported correctly
- Fix bug where `traffic-manager endpoint update` did not work
- Add 'network watcher' preview commands

Profile

- Support login when there are no subscriptions found ([#2560 ↗](#))
- Support short param name in az account set --subscription ([#2980 ↗](#))

Redis

- Adding update command which also adds the ability to scale for redis cache
- Deprecates the 'update-settings' command

Resource

- Add managedapp and managedapp definition commands ([#2985 ↗](#))
- Support 'provider operation' commands ([#2908 ↗](#))
- Support generic resource create ([#2606 ↗](#))
- Fix resource parsing and api version lookup. ([#2781 ↗](#))
- Add docs for az lock update. ([#2702 ↗](#))
- Error out if you try to list resources for a group that doesn't exist. ([#2769 ↗](#))
- [Compute] Fix issues with VMSS and VM availability set update. ([#2773 ↗](#))
- Fix lock create and delete if parent-resource-path is None ([#2742 ↗](#))

Role

- create-for-rbac: ensure SP's end date will not exceed certificate's expiration date ([#2989 ↗](#))
- RBAC: add full support for 'ad group' ([#2016 ↗](#))
- role: fix issues on role definition update ([#2745 ↗](#))
- create-for-rbac: ensure user provided password is picked up

SQL

- Added az sql server list-usages and az sql db list-usages commands
- SQL - ability to connect directly to resource provider ([#2832 ↗](#))

Storage

- Default location to resource group location for `storage account create`
- Add support for incremental blob copy
- Add support for large block blob upload
- Change block size to 100MB when file to upload is larger than 200GB

VM

- avail-set: make UD&FD domain counts optional
 - note: VM commands in sovereign clouds Please avoid managed disk related features, including the following:
 1. az disk/snapshot/image
 2. az vm/vmss disk
 3. Inside "az vm/vmss create", use "—use-unmanaged-disk" to avoid managed disk Other commands should work
- vm/vmss: improve the warning text when generates ssh key pairs
- vm/vmss: support create from a market place image which requires plan info ([#1209 ↗](#))

April 3, 2017

Version 2.0.2

We released the ACR, Batch, KeyVault, and SQL components in this release

text

```
azure-cli (2.0.2)

acr (2.0.0)
acs (2.0.2)
appservice (0.1.2)
batch (2.0.0)
cloud (2.0.0)
component (2.0.0)
configure (2.0.2)
container (0.1.2)
core (2.0.2)
documentdb (0.1.2)
feedback (2.0.0)
find (0.0.1b1)
iot (0.1.2)
keyvault (2.0.0)
lab (0.0.1)
monitor (0.0.1)
network (2.0.2)
nspkg (2.0.0)
profile (2.0.2)
redis (0.1.1b3)
resource (2.0.2)
role (2.0.1)
sql (2.0.0)
storage (2.0.2)
vm (2.0.2)
```

Core

- Add acr, lab, monitor, and find modules to default list
- Login: skip erroneous tenant ([#2634 ↗](#))
- login: set default subscription to one with the state of "Enabled" ([#2575 ↗](#))
- Add wait commands and --no-wait support to more commands ([#2524 ↗](#))
- core: support login using service principal with a cert ([#2457 ↗](#))
- Add prompting for missing template parameters. ([#2364 ↗](#))
- Support setting default values for common arguments like default resource group, default web, default vm
- Support login to specific tenant

ACS

- [ACS] Adding support for configuring a default ACS cluster ([#2554 ↗](#))
- Add support for ssh key password prompting. ([#2044 ↗](#))
- Add support for windows clusters. ([#2211 ↗](#))

- Switch from Owner to Contributor role. ([#2321 ↗](#))

AppService

- appservice: support to get external ip address used for DNS A records ([#2627 ↗](#))
- appservice: support binding wildcard certificates ([#2625 ↗](#))
- appservice: support list publishing profiles ([#2504 ↗](#))
- AppService - Trigger source control sync after config ([#2326 ↗](#))

DataLake

- Initial release of Data Lake Analytics module
- Initial release of Data Lake Store module

DocuemntDB

- DocumentDB: Adding support for listing connection strings ([#2580 ↗](#))

VM

- [Compute] Add AppGateway support to virtual machine scale set create ([#2570 ↗](#))
- [VM/VMSS] Improved disk caching support ([#2522 ↗](#))
- VM/VMSS: incorporate credentials validation logic used by portal ([#2537 ↗](#))
- Add wait commands and --no-wait support ([#2524 ↗](#))
- Virtual machine scale set: support * to list instance view across vms ([#2467 ↗](#))
- Add --secrets for VM and virtual machine scale set ([#2212]
(<https://github.com/Azure/azure-cli/pull/2212>))
- Allow VM creation with specialized VHD ([#2256 ↗](#))

February 27, 2017

Version 2.0.0

This release of Azure CLI 2.0 is the first "Generally Available" release General availability applies to these command modules:

- Container Service (acs)
- Compute (including Resource Manager, VM, virtual machine scale sets, Managed Disks)
- Networking
- Storage

These command modules can be used in production and are supported by standard Microsoft SLA You can open issues directly with Microsoft support or on our [github issues list](#) You can ask questions on [StackOverflow using the azure-cli tag](#), or contact the product team at azfeedback@microsoft.com You can provide feedback from the command line with the `az feedback` command

The commands in these modules are stable and the syntax is not expected to change in upcoming releases of this version of Azure CLI

To verify the version of the CLI, use `az --version` The output lists the version of the CLI itself (2.0.0 in this release), the individual command modules, and the versions of Python and GCC that you're using

text

```
azure-cli (2.0.0)

acs (2.0.0)
appservice (0.1.1b5)
batch (0.1.1b4)
cloud (2.0.0)
component (2.0.0)
configure (2.0.0)
container (0.1.1b4)
core (2.0.0)
documentdb (0.1.1b2)
feedback (2.0.0)
iot (0.1.1b3)
keyvault (0.1.1b5)
network (2.0.0)
nspkg (2.0.0)
profile (2.0.0)
redis (0.1.1b3)
resource (2.0.0)
role (2.0.0)
sql (0.1.1b5)
storage (2.0.0)
vm (2.0.0)

Python (Darwin) 2.7.10 (default, Jul 30 2016, 19:40:32)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.34)]
```

Note

Some of the command modules have a "bn" or "rcn" postfix These command modules are still in preview and will become generally available in the future

We also have nightly preview builds of the CLI. For information, see these instructions on [getting the nightly builds](#), and these instructions on [developer setup and contributing code](#).

You can report issues with nightly preview builds in the following ways:

- Report issues in our [github issues list](#)
- Contact the product team at azfeedback@microsoft.com
- Provide feedback from the command line with the `az feedback` command

MSAL-based Azure CLI

Article • 01/23/2024

Starting in version 2.30.0, Azure CLI uses Microsoft Authentication Library ([MSAL](#)) as the underlying authentication library. MSAL uses Azure Active Directory v2.0 authentication flow to provide more functionality and increases security for token cache.

⚠️ Warning

BREAKING CHANGES are introduced in Azure CLI 2.30.0. Carefully read document prior to installation.

accessTokens.json deprecation

Previous versions of Azure CLI saved Azure AD Authentication Library (ADAL) tokens and service principal entries to `~/.azure/accessToken.json`. The latest versions of the Azure CLI use MSAL and no longer generate `accessTokens.json`. Any existing workflow depending on `accessTokens.json` no longer works.

The MSAL token cache and service principal entries are saved as encrypted files on Windows, and plaintext files on Linux and macOS.

ⓘ Important

When using Azure CLI in a pipeline like Azure DevOps, ensure all tasks and stages are using versions of Azure CLI above v2.30.0 for MSAL-based Azure CLI. Azure CLI 2.30.0 is not backwards compatible with prior versions and throws an error when working with versions prior to 2.30.0.

Alternatives to consider

Alternatives you may consider for stability:

Calling `az account get-access-token`

You can manually call `az account get-access-token` in a terminal or use a subprocess to call it from another programming language. By default, the returned access token is for

Azure Resource Manager (ARM) and the default subscription/tenant shown in [az account show](#).

```
Azure CLI

# get the active subscription
az account show --output table

# get access token for the active subscription
az account get-access-token

# get access token for a specific subscription
az account get-access-token --subscription "<subscription ID or name>"
```

To learn more about the access tokens, see [Sign in with Azure CLI](#).

Using `AzureCliCredential`

`AzureCliCredential` is a credential type in all existing language SDKs. It uses subprocess to call `az account get-access-token` to get an access token for the current logged-in account.

See also

- MSAL
 - [Overview of the Microsoft Authentication Library \(MSAL\)](#)
 - [Migrate applications to the Microsoft Authentication Library \(MSAL\)](#)
- Python
 - [AzureCliCredential Class in Python](#)
- .NET
 - [AzureCliCredential Class in .NET](#)
- Java
 - [AzureCliCredential Class in Java](#)

Microsoft Graph migration

Article • 08/02/2023

Due to [the deprecation of Azure Active Directory \(Azure AD\) Graph](#), the underlying Active Directory Graph API replaces [Microsoft Graph API](#) in Azure CLI 2.37.0.

Breaking changes

For differences of the underlying API and output JSON breaking changes, refer to [Property differences between Azure AD Graph and Microsoft Graph](#).

For example, the most outstanding change is that `id` replaces the `objectId` property in the output JSON of a Graph object.

Command argument and behavior breaking changes are listed in the next section.

`az ad app create/update`

- Split `--reply-urls` into `--web-redirect-uris` and `--public-client-redirect-uris`
- Replace `--homepage` with `--web-home-page-url`
- Replace `--available-to-other-tenants` with `--sign-in-audience`
- Replace `--native-app` with `--is-fallback-public-client`
- Replace `--oauth2-allow-implicit-flow` with `--enable-access-token-issuance`
- Add `--enable-id-token-issuance` to set
`web/implicitGrantSettings/enableIdTokenIssuance`
- Remove `--password` and `--credential-description`. Use `az ad app credential reset` to let Graph service create a password for you (<https://github.com/Azure/azure-cli/issues/20675>)
- Add `--key-display-name` to set `keyCredential`'s `displayName`

`az ad app permission grant`

- Remove `--expires`
- `--scope` no longer defaults to `user_impersonation` and is now required

`az ad app credential reset`

- Replace `--credential-description` with `--display-name`
(<https://github.com/Azure/azure-cli/issues/20561>)

- Remove `--password`. Without specifying certificate arguments, Graph service creates a password for you (<https://github.com/Azure/azure-cli/issues/20675>)

az ad sp delete

- This command no longer deletes the corresponding application. Use `az ad app delete` to explicitly delete the application (<https://github.com/Azure/azure-cli/issues/8467>)
- This command no longer deletes corresponding role assignments of the service principal. Use `az role assignment delete` to explicitly delete role assignments (<https://github.com/Azure/azure-cli/issues/20805>)

az ad sp credential

- This command group now operates on service principal, not application (<https://github.com/Azure/azure-cli/issues/11458>)

az ad sp credential reset

- Replace `--name` with `--id`
- Remove `--password`. Without specifying certificate arguments, Graph service creates a password for you (<https://github.com/Azure/azure-cli/issues/20675>)

az ad user create

- Replace `--force-change-password-next-login` with `--force-change-password-next-sign-in`

az ad user update

- Replace `--force-change-password-next-login` with `--force-change-password-next-sign-in`

az ad group get-member-groups

- Remove `--additional-properties`

az ad group member add

- Remove `--additional-properties`

Known issues

- Regarding generic update arguments, the only supported operation is using `--set` on the root level of a Graph object. When you use `--add`, `--remove` or `--set` on sublevels currently doesn't work (due to the underlying infrastructure change). For unsupported scenarios, you may use `az rest` to directly call [Microsoft Graph API](#). Examples can be found at <https://github.com/Azure/azure-cli/issues/22580>.
- Microsoft Graph related commands like `az ad` and `az role` fail in Azure Stack environments that don't have Microsoft Graph support. Use Azure CLI 2.36.0 or earlier versions for Azure Stack environments.

Install a previous version

If you aren't ready for the migration yet, such as lacking Microsoft Graph permissions, you may keep using Azure CLI versions \leq 2.36.0. If you have already installed 2.37.0, you may roll back to a previous version following the "Install specific version" section under the [installation documents](#) (except for Homebrew, which doesn't support installing previous versions).

Troubleshooting

Graph command fails with `AADSTS50005` or `AADSTS53000`

Your tenant may have Conditional Access policies that block using device code flow to access Microsoft Graph. In such cases, use authorization code flow or a service principal to sign in instead. For more information about sign in methods, see [Sign in with Azure CLI](#).

Microsoft tenant (72f988bf-86f1-41af-91ab-2d7cd011db47) has such Conditional Access policies configured.

More information

You can find more information about the Microsoft Graph migration on <https://github.com/Azure/azure-cli/issues/22580>.

Give feedback

If you have any questions, reply to <https://github.com/Azure/azure-cli/issues/22580> or create a new issue with the `az feedback` command.

Azure CLI conceptual article list

Article • 12/04/2023

This article provides an A - Z list of published content for each Azure CLI reference group. Use these links to find tested code snippets and scripts that show the use of parameter values for different tasks. For common Azure CLI commands, like `az group`, only the most popular articles have been listed.

With over 9,000 reference commands, not every Azure CLI command has been used in Azure published content. For a complete list of Azure CLI commands, see the [A - Z reference list](#). For a list of Azure CLI samples, see [Azure CLI sample scripts](#)

az account

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az account	How to manage Azure subscriptions with the Azure CLI
	Get started with Azure CLI
	Create an Azure service principal with Azure CLI
	Sign in interactively with Azure CLI
	Tips for using the Azure CLI successfully
	Output formats for Azure CLI commands
	Learn to use Bash with the Azure CLI
	How to use variables in Azure CLI commands
	How to manage Azure resource groups with the Azure CLI
	Prepare your environment for the Azure CLI
az account alias	Programmatically create Azure Enterprise Agreement subscriptions with the latest APIs
	Programmatically create Azure subscriptions for a Microsoft Customer Agreement with the latest APIs
	Programmatically create Azure subscriptions for a Microsoft Partner Agreement with the latest APIs

Reference subgroup	Azure CLI article showing reference use
az account lock	How to manage Azure subscriptions with the Azure CLI
az account management-group	How to manage Azure subscriptions with the Azure CLI
	Assign Azure roles using Azure CLI
	Manage your Azure subscriptions at scale with management groups
	List Azure role assignments using Azure CLI
	Quickstart: Create a management group with the Azure CLI
az account management-group subscription	How to manage Azure subscriptions with the Azure CLI
	Manage your Azure subscriptions at scale with management groups
az account subscription	Configure your local JavaScript dev environment for Azure
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
az account tenant	How to manage Azure subscriptions with the Azure CLI
	How to find your Microsoft Entra tenant ID
	Move a Log Analytics workspace to a different subscription or resource group

az acr

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az acr	Authenticate with an Azure container registry
	Push your first image to your Azure container registry using the Docker CLI
	Azure Container Registry roles and permissions
	Configure a custom container for Azure App Service
	Working with containers and Azure Functions

Reference subgroup	Azure CLI article showing reference use
	Build and push Docker images to Azure Container Registry using Docker templates
	Authenticate with Azure Container Registry (ACR) from Azure Kubernetes Service (AKS)
	Create an unmanaged ingress controller
	Migrate custom software to Azure App Service using a custom container
	Tutorial: Deploy a multi-container group using Docker Compose
az acr agentpool	Run an ACR task on a dedicated agent pool
az acr artifact-streaming	Artifact streaming in Azure Container Registry (Preview)
	Reduce image pull time with Artifact Streaming on Azure Kubernetes Service (AKS) (Preview)
az acr artifact-streaming operation	Artifact streaming in Azure Container Registry (Preview)
az acr cache	Enable Artifact Cache - Azure CLI
	Enable Artifact Cache with authentication - Azure CLI
az acr config authentication-as-arm	Azure Container Apps image pull with managed identity
	Disable authentication as ARM template
az acr config content-trust	Content trust in Azure Container Registry
az acr config retention	Set a retention policy for untagged manifests
az acr config soft-delete	Recover deleted artifacts with soft delete policy in Azure Container Registry (Preview)
az acr connected-registry	Pull images from a connected registry on IoT Edge device
	Quickstart: Deploy a connected registry to an IoT Edge device
	Quickstart: Create a connected registry using the Azure CLI
	Tutorial: Deploy a connected registry to a nested IoT Edge hierarchy
az acr credential	Migrate custom software to Azure App Service using a custom container

Reference subgroup	Azure CLI article showing reference use
	Deploy a containerized Flask or FastAPI web app on Azure App Service
	Create your first containerized functions on Azure Container Apps
	Create your first containerized Azure Functions
	Quickstart: Use Azure App Configuration in Azure Container Apps
	Build and deploy a Python web app with Azure Container Apps and PostgreSQL
	Use Azure Container Registry with Azure Red Hat OpenShift (ARO)
	Quickstart: Deploy an existing container image with the command line
	Create your first containerized Azure Functions on Azure Arc (preview)
	Deploy a Go web app to Azure Container Apps
az acr credential-set	Enable Artifact Cache with authentication - Azure CLI
az acr encryption	Enable a customer-managed key
	Rotate and revoke a customer-managed key
az acr export-pipeline	ACR Transfer with Az CLI
az acr helm	Azure Container Registry webhook reference
az acr identity	Troubleshoot a customer-managed key
az acr import-pipeline	ACR Transfer with Az CLI
az acr manifest	Push and pull Helm charts to an Azure container registry
	Import container images to a container registry
	Create a token with repository-scoped permissions
	About registries, repositories, and artifacts
	Delete container images in Azure Container Registry
	Artifact streaming in Azure Container Registry (Preview)
	Recover deleted artifacts with soft delete policy in Azure Container Registry (Preview)
	Set a retention policy for untagged manifests

Reference subgroup	Azure CLI article showing reference use
	Reduce image pull time with Artifact Streaming on Azure Kubernetes Service (AKS) (Preview)
	Lock a container image in an Azure container registry
<code>az acr network-rule</code>	Connect privately to an Azure container registry using Azure Private Link
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Configure public IP network rules
	Restrict access to a container registry using a service endpoint in an Azure virtual network
<code>az acr pack</code>	Build and push an image from an app using a Cloud Native Buildpack
<code>az acr pipeline-run</code>	ACR Transfer with Az CLI
<code>az acr private-endpoint-connection</code>	Troubleshooting online endpoints deployment and scoring
	Connect privately to an Azure container registry using Azure Private Link
	Secure your managed online endpoints with network isolation
<code>az acr replication</code>	Geo-replication in Azure Container Registry
	Enable zone redundancy in Azure Container Registry for resiliency and high availability
<code>az acr repository</code>	Push your first image to your Azure container registry using the Docker CLI
	Troubleshooting online endpoints deployment and scoring
	Tutorial: Deploy a multi-container group using Docker Compose
	Push and pull Helm charts to an Azure container registry
	Tutorial - Create an Azure Container Registry (ACR) and build images
	Create a token with repository-scoped permissions
	Tutorial: Create an Azure container registry and push a container image
	Quickstart: Create a private container registry using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Delete container images in Azure Container Registry
	Recover deleted artifacts with soft delete policy in Azure Container Registry (Preview)
az acr scope-map	Create a token with repository-scoped permissions
	Pull images from a connected registry on IoT Edge device
az acr task	Automatically purge images from an Azure container registry
	ACR Tasks reference: YAML
	Tutorial: Automate container image builds in the cloud when you commit source code
	Tutorial: Automate container image builds when a base image is updated in an Azure container registry
	Tutorial: Run an ACR task on a defined schedule
	Run an ACR task on a dedicated agent pool
	Use an Azure-managed identity in ACR Tasks
	About base image updates for ACR Tasks
	Tutorial: Run a multi-step container workflow in the cloud when you commit source code
	Cross-registry authentication in an ACR task using an Azure-managed identity
az acr task credential	Use an Azure-managed identity in ACR Tasks
	Tutorial: Run a multi-step container workflow in the cloud when you commit source code
	Cross-registry authentication in an ACR task using an Azure-managed identity
	How to consume and maintain public content with Azure Container Registry Tasks
	Tutorial: Automate container image builds when a base image is updated in another private container registry
az acr task timer	Tutorial: Run an ACR task on a defined schedule
az acr token	Push and pull Helm charts to an Azure container registry

Reference subgroup	Azure CLI article showing reference use
	Create a token with repository-scoped permissions
	Pull images from a connected registry on IoT Edge device
	How to consume and maintain public content with Azure Container Registry Tasks
az acr token credential	Create a token with repository-scoped permissions
	Pull images from a connected registry on IoT Edge device
az acr webhook	Migrate custom software to Azure App Service using a custom container
	Continuous deployment with custom containers in Azure App Service
	Using Azure Container Registry webhooks

az ad

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az ad app	Tips for using the Azure CLI successfully
	Considerations for running the Azure CLI in a PowerShell environment
	Deploy to App Service using GitHub Actions
	Provision and publish a bot
	Tutorial: Authenticate and authorize users end-to-end in Azure App Service
	Configure a GitHub Action to create a container instance
	Tutorial: Use GitHub Actions to deploy to App Service and connect to a database
	Integrate Microsoft Entra ID with Azure Kubernetes Service (AKS) using the Azure CLI (legacy)
	Use GitHub Actions workflow to deploy your static website in Azure Storage
	Quickstart: Deploy Bicep files by using GitHub Actions

Reference subgroup	Azure CLI article showing reference use
az ad app credential	<p>Provision and publish a bot</p> <p>Update or rotate the credentials for an Azure Kubernetes Service (AKS) cluster</p> <p>Use a service principal with Azure Kubernetes Service (AKS)</p> <p>Use SAP Deployment Automation Framework from Azure DevOps Services</p> <p>Rotate service principal credentials for your Azure Red Hat OpenShift (ARO) Cluster</p> <p>Tutorial: Using automation to set up the Microsoft Entra admin for SQL Server</p> <p>Tutorial: Enterprise scale for SAP Deployment Automation Framework</p> <p>Create an Immersive Reader resource and configure Microsoft Entra authentication</p> <p>Use Azure Key Vault to deliver TLS/SSL certificates to the JVM</p> <p>Create an app registration to use with Azure Digital Twins</p>
az ad app federated-credential	<p>Deploy to App Service using GitHub Actions</p> <p>Configure an app to trust an external identity provider</p> <p>Configure a GitHub Action to create a container instance</p> <p>Tutorial: Use GitHub Actions to deploy to App Service and connect to a database</p> <p>Use GitHub Actions workflow to deploy your static website in Azure Storage</p> <p>Use GitHub Actions to connect to Azure</p> <p>Deploy ARM templates by using GitHub Actions</p> <p>Use GitHub Actions to connect to Azure SQL Database</p> <p>Use GitHub Actions with Azure Machine Learning</p> <p>Configure cross-tenant customer-managed keys for an existing storage account</p>
az ad app permission	<p>Integrate Microsoft Entra ID with Azure Kubernetes Service (AKS) using the Azure CLI (legacy)</p>

Reference subgroup	Azure CLI article showing reference use
	Use Azure RBAC on Azure Arc-enabled Kubernetes clusters (preview)
	Quickstart: Deploy RESTful API application to Azure Spring Apps
	Tutorial: Using automation to set up the Microsoft Entra admin for SQL Server
	Configure Microsoft Entra authentication for an Azure Red Hat OpenShift 4 cluster (CLI)
az ad group	Assign a Key Vault access policy (legacy)
	Deploy Azure Virtual Desktop
	Enable Azure managed identity authentication for Kubernetes clusters with kubelogin
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Use Kubernetes role-based access control with Microsoft Entra ID in Azure Kubernetes Service
	Assign Azure roles using Azure CLI
	Use Azure role-based access control to define access to the Kubernetes configuration file in Azure Kubernetes Service (AKS)
	Assign Azure roles using Azure Resource Manager templates
	Authenticate .NET apps to Azure services during local development using developer accounts
az ad group member	Assign a Key Vault access policy (legacy)
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Use Kubernetes role-based access control with Microsoft Entra ID in Azure Kubernetes Service
	Authenticate .NET apps to Azure services during local development using developer accounts

Reference subgroup	Azure CLI article showing reference use
	Authenticate .NET apps to Azure services during local development using service principals
	Authenticate Python apps to Azure services during local development using service principals
	Authenticate Python apps to Azure services during local development using developer accounts
	Authenticate JavaScript apps to Azure services during local development using service principals
	Authenticate JavaScript apps to Azure services during local development using developer accounts
az ad signed-in-user	Quickstart: Create, download, and list blobs with Azure CLI
	Use Microsoft Entra ID for authentication with Azure Database for PostgreSQL - Flexible Server
	Tutorial: Access Azure services from a .NET web app
	Access Azure Storage from a web app using managed identities
	Add Key Management Service etcd encryption to an Azure Kubernetes Service cluster
	Integrate Microsoft Entra ID with Azure Kubernetes Service (AKS) using the Azure CLI (legacy)
	Sign container images with Notation and Azure Key Vault using a self-signed certificate
	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments
	Use cluster connect to securely connect to Azure Arc-enabled Kubernetes clusters
	Quickstart: Provision and activate a Managed HSM using Azure CLI
az ad sp	Create an Azure service principal with Azure CLI
	Get an existing service principal
	Use an Azure service principal with certificate-based authentication
	Use an Azure service principal with password-based authentication
	Reset service principal credentials

Reference subgroup	Azure CLI article showing reference use
	Cleanup & troubleshoot service principals
	Application and service principal objects in Microsoft Entra ID
	Assign a Key Vault access policy (legacy)
	Transfer an Azure subscription to a different Microsoft Entra directory
	Access control lists (ACLs) in Azure Data Lake Storage Gen2
az ad sp credential	Create an Azure service principal with Azure CLI Output formats for Azure CLI commands Use an Azure service principal with certificate-based authentication Reset service principal credentials
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Azure Container Registry authentication with service principals
	Pull images from an Azure container registry to a Kubernetes cluster using a pull secret
	Integrate Microsoft Entra ID with Azure Kubernetes Service (AKS) using the Azure CLI (legacy)
	Use Azure RBAC on Azure Arc-enabled Kubernetes clusters (preview)
	Rotate service principal credentials for your Azure Red Hat OpenShift (ARO) Cluster
az ad user	Log in to a Windows virtual machine in Azure by using Microsoft Entra ID including passwordless Assign a Key Vault access policy (legacy)
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	Create a virtual network peering - Resource Manager, different subscriptions and Microsoft Entra tenants
	Tutorial: Connect a function app to Azure SQL with managed identity and SQL bindings

Reference subgroup	Azure CLI article showing reference use
	Use Kubernetes role-based access control with Microsoft Entra ID in Azure Kubernetes Service
	Assign Azure roles using Azure CLI
	Use Azure role-based access control to define access to the Kubernetes configuration file in Azure Kubernetes Service (AKS)
	Assign Azure roles using Azure Resource Manager templates

az afd

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az afd endpoint	Endpoints in Azure Front Door
	Configure a Web Application Firewall rate-limit rule
	Tutorial: Create a highly available multi-region app in Azure App Service
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Cache purging in Azure Front Door with Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd origin	Tutorial: Create a highly available multi-region app in Azure App Service
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Connect Azure Front Door Premium to an App Service origin with Private Link using Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
	Connect Azure Front Door Premium to a Storage Account origin with Private Link with Azure CLI
az afd origin-group	Tutorial: Create a highly available multi-region app in Azure App Service
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd profile	Configure a Web Application Firewall rate-limit rule
	Tutorial: Create a highly available multi-region app in Azure App Service
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd route	Tutorial: Create a highly available multi-region app in Azure App Service
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd rule	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd rule action	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd rule condition	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd rule-set	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI
az afd security-policy	Configure a Web Application Firewall rate-limit rule
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front Door Standard/Premium (Preview) with Azure CLI

az aks

Expand table

Reference subgroup	Azure CLI article showing reference use
az aks	Supported Kubernetes versions in Azure Kubernetes Service (AKS)

Reference subgroup	Azure CLI article showing reference use
	Use the Azure Key Vault provider for Secrets Store CSI Driver in an Azure Kubernetes Service (AKS) cluster
	Quickstart: Deploy an Azure Kubernetes Service (AKS) cluster using Azure CLI
	Create a private Azure Kubernetes Service (AKS) cluster
	Use a managed identity in Azure Kubernetes Service (AKS)
	Run a self-hosted agent in Docker
	Enable Azure managed identity authentication for Kubernetes clusters with kubelogin
	Connect to Azure Kubernetes Service (AKS) cluster nodes for maintenance or troubleshooting
	Quickstart: Deploy an Azure Kubernetes Service (AKS) cluster using Azure portal
	Managed NGINX ingress with the application routing add-on
az aks addon	Azure Key Vault provider for Secrets Store CSI Driver for Azure Kubernetes Service (AKS) configuration and troubleshooting options
	Plug in CA certificates for Istio-based service mesh add-on on Azure Kubernetes Service
	Confidential computing plugin for Confidential VMs
az aks approuting	Managed NGINX ingress with the application routing add-on
	Set up a custom domain name and SSL certificate with the application routing add-on
	Migrate from HTTP application routing to the application routing add-on
az aks approuting zone	Set up a custom domain name and SSL certificate with the application routing add-on
	Configure NGINX ingress controller to support Azure private DNS zone with application routing add-on
az aks command	Access a private Azure Kubernetes Service (AKS) cluster
az aks draft	Draft for Azure Kubernetes Service (AKS) (preview)

Reference subgroup	Azure CLI article showing reference use
az aks egress-endpoints	Troubleshoot connections to endpoints outside the virtual network
az aks machine	Connect to Azure Kubernetes Service (AKS) cluster nodes for maintenance or troubleshooting
az aks maintenanceconfiguration	Use Planned Maintenance to schedule and control upgrades for your Azure Kubernetes Service (AKS) cluster
	Azure Kubernetes Service patch and upgrade guidance
az aks mesh	Deploy Istio-based service mesh add-on for Azure Kubernetes Service
	Azure Kubernetes Service (AKS) external or internal ingresses for Istio service mesh add-on deployment
	Upgrade Istio-based service mesh add-on for Azure Kubernetes Service
	Plug in CA certificates for Istio-based service mesh add-on on Azure Kubernetes Service
az aks mesh upgrade	Upgrade Istio-based service mesh add-on for Azure Kubernetes Service
az aks nodepool	Use a managed identity in Azure Kubernetes Service (AKS)
	Use the cluster autoscaler in Azure Kubernetes Service (AKS)
	Configure Azure CNI Overlay networking in Azure Kubernetes Service (AKS)
	Create node pools for a cluster in Azure Kubernetes Service (AKS)
	Upgrade an Azure Kubernetes Service (AKS) cluster
	Configure an AKS cluster
	Manage system node pools in Azure Kubernetes Service (AKS)
	Upgrade Azure Kubernetes Service (AKS) node images
	Secure traffic between pods using network policies in Azure Kubernetes Service (AKS)
	Resize node pools in Azure Kubernetes Service (AKS)
az aks nodepool snapshot	Azure Kubernetes Service (AKS) node pool snapshot

Reference subgroup	Azure CLI article showing reference use
az aks oidc-issuer	Create an OpenID Connect provider on Azure Kubernetes Service (AKS)
az aks pod-identity	Use Microsoft Entra pod-managed identities in Azure Kubernetes Service (Preview)
	Migrate from pod managed-identity to workload identity
az aks pod-identity exception	Use Microsoft Entra pod-managed identities in Azure Kubernetes Service (Preview)
	Troubleshoot Azure Kubernetes Service backup and restore
az aks trustedaccess rolebinding	Manage Azure Kubernetes Service backups using Azure Backup
	Get secure access for Azure resources in Azure Kubernetes Service by using Trusted Access (preview)
	Back up Azure Kubernetes Service using Azure CLI
	Restore Azure Kubernetes Service using Azure CLI

az aksarc

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az aksarc	Create Kubernetes clusters using Azure CLI
	Azure Hybrid Benefit for AKS enabled by Azure Arc (AKS on Azure Stack HCI 23H2)
	Use GPUs for compute-intensive workloads (AKS on Azure Stack HCI 23H2)
	Use Container Storage Interface (CSI) file drivers in AKS enabled by Azure Arc
	Upgrade an Azure Kubernetes Service (AKS) cluster
	Manage node pools for a cluster (AKS on Azure Stack HCI 23H2)
	Quickstart: deploy a Kubernetes cluster using an Azure Resource Manager template
	Retrieve certificate-based admin kubeconfig

Reference subgroup	Azure CLI article showing reference use
	Manage control plane nodes in a Kubernetes cluster
	Use cluster labels in AKS enabled by Azure Arc
az aksarc logs	Get on-demand logs for troubleshooting
az aksarc nodepool	Use GPUs for compute-intensive workloads (AKS on Azure Stack HCI 23H2)
	Manage node pools for a cluster (AKS on Azure Stack HCI 23H2)
	Create Windows Server containers
	Use cluster labels in AKS enabled by Azure Arc
az aksarc release	Use manual (offline) download in AKS enabled by Azure Arc
az aksarc vmsize	Use GPUs for compute-intensive workloads (AKS on Azure Stack HCI 23H2)

az akshybrid

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az akshybrid	Before you begin: uninstall the AKS cluster provisioning preview
az akshybrid nodepool	Create Windows Server containers
az akshybrid vnet	Before you begin: uninstall the AKS cluster provisioning preview

az alerts-management

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az alerts-management prometheus-rule-group	Azure Monitor managed service for Prometheus rule groups

az alias

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az alias	How to use the Azure CLI alias extension

az ams

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az ams account	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI
	Tutorial: Give an Azure Function app access to a Media Services account
az ams account sp	Azure CLI example: Reset the account credentials
az ams account-filter	Creating filters with CLI
az ams asset	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI
az ams asset-filter	Creating filters with CLI
az ams job	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI
az ams live-event	Tutorial: Give an Azure Function app access to a Media Services account
az ams streaming-endpoint	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI
az ams streaming-locator	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI
	Creating filters with CLI
az ams transform	Tutorial: Encode a remote file based on URL and stream the video - Azure CLI

az aosm

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az aosm nfd	Quickstart: Complete the prerequisites to deploy a Containerized Network Function in Azure Operator Service Manager
	Quickstart: Publish Nginx container as Containerized Network Function (CNF)
	Use Azure Operator Service Manager (AOSM) CLI extension
	Quickstart: Publish Ubuntu Virtual Machine (VM) as Virtual Network Function (VNF)
az aosm nsd	Use Azure Operator Service Manager (AOSM) CLI extension
	Quickstart: Design a Containerized Network Function (CNF) Network Service Design with Nginx
	Quickstart: Design a Network Service Design (NSD) for Ubuntu Virtual Machine (VM) as a Virtualized Network Function (VNF)

az apic

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az apic api	Use the Azure CLI to manage your API inventory
az apic api definition	Use the Azure CLI to manage your API inventory
az apic api version	Use the Azure CLI to manage your API inventory
az apic service	Import APIs to your API center from Azure API Management

az apim

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az apim	Migrate an API Management instance hosted on the stv1 platform to stv2
	Deploy an Azure API Management instance to multiple Azure regions
	Quickstart: Create an Azure API Management instance using Terraform

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create a new Azure API Management instance by using the Azure CLI
	Import APIs to your API center from Azure API Management
az apim api	Tutorial: Create and publish a product Import an OpenAPI specification Import a GraphQL API Import SOAP API to API Management Tutorial: Use revisions to make non-breaking API changes safely
az apim api operation	Tutorial: Mock API responses
az apim api release	Tutorial: Use revisions to make non-breaking API changes safely
az apim api versionset	Tutorial: Publish multiple versions of your API
az apim nv	Use named values in Azure API Management policies
az apim product	Tutorial: Create and publish a product
az apim product api	Tutorial: Create and publish a product

az appconfig

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az appconfig	Quickstart: Create an Azure App Configuration store Disable public access in Azure App Configuration Use customer-managed keys to encrypt your App Configuration data Set up private access in Azure App Configuration Tutorial: Connect a web app to Azure App Configuration with Service Connector How to use managed identities for Azure App Configuration

Reference subgroup	Azure CLI article showing reference use
	App configuration support
	Use Event Grid for App Configuration data change notifications
	Create an Azure App Configuration store with the Azure CLI
	Delete an Azure App Configuration store with the Azure CLI
az appconfig credential	Disable access key authentication for an Azure App Configuration instance
	App configuration support
	Create an Azure App Configuration store with the Azure CLI
az appconfig feature	Quickstart: Create an Azure App Configuration store
az appconfig identity	Use customer-managed keys to encrypt your App Configuration data
	How to use managed identities for Azure App Configuration
az appconfig kv	Quickstart: Create an Azure App Configuration store
	Use content type to store JSON key-values in App Configuration
	Import or export configuration data
	Azure App Configuration support for configuration files
	Point-in-time key-values
	Tutorial: Connect a web app to Azure App Configuration with Service Connector
	Work with key-values in an Azure App Configuration store
	Integrate with Kubernetes Deployment using Helm
	Integrate with a CI/CD pipeline
	App configuration support
az appconfig replica	Enable geo-replication
	App configuration support
az appconfig revision	Point-in-time key-values

az appservice

Reference subgroup	Azure CLI article showing reference use
az appservice	Tutorial: Host a RESTful API with CORS in Azure App Service Create a PHP web app in Azure App Service
	Configure Premium V3 tier for Azure App Service
	Tutorial: Create a multi-container (preview) app in Web App for Containers
	Create a multi-container (preview) app using a Docker Compose configuration
	Create an Angular app with Azure Cosmos DB's API for MongoDB - Create an Azure Cosmos DB account
az appservice ase	Using Private Endpoints for App Service apps App Service Environment networking
	Use the in-place migration feature to migrate App Service Environment v1 and v2 to App Service Environment v3
	Network configuration settings
	Upgrade preference for App Service Environment planned maintenance
	Use the side by side migration feature to migrate App Service Environment v2 to App Service Environment v3 (Preview)
az appservice kube	Set up an Azure Arc-enabled Kubernetes cluster to run App Service, Functions, and Logic Apps (Preview)
az appservice plan	Manage your function app Tutorial: Host a RESTful API with CORS in Azure App Service
	Migrate custom software to Azure App Service using a custom container
	Reliability in Azure App Service
	Tutorial: Create a multistage pipeline with Azure DevOps
	Tutorial: Use a managed identity to connect Key Vault to an Azure web app in .NET
	Automatic scaling in Azure App Service
	Create a PHP web app in Azure App Service

Reference subgroup	Azure CLI article showing reference use
	Deploy a containerized Flask or FastAPI web app on Azure App Service
	Configure Premium V3 tier for Azure App Service

az arcappliance

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az arcappliance	Troubleshoot Azure Arc resource bridge issues
	Upgrade Arc resource bridge
	Perform ongoing administration for Arc-enabled VMware vSphere
	Perform ongoing administration for Arc-enabled System Center Virtual Machine Manager
az arcappliance delete	Before you begin: uninstall the AKS cluster provisioning preview
az arcappliance logs	Perform ongoing administration for Arc-enabled VMware vSphere
	Perform ongoing administration for Arc-enabled System Center Virtual Machine Manager
az arcappliance prepare	Troubleshoot Azure Arc resource bridge issues
az arcappliance update-infracredentials	Perform ongoing administration for Arc-enabled VMware vSphere
	Perform ongoing administration for Arc-enabled System Center Virtual Machine Manager
az arcappliance upgrade	Upgrade Arc resource bridge
	Perform ongoing administration for Arc-enabled VMware vSphere
	Perform ongoing administration for Arc-enabled System Center Virtual Machine Manager

az arcdat

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az arcdata ad-connector	Tutorial - Deploy Active Directory connector using Azure CLI
az arcdata dc	Uninstall Azure Arc-enabled data services Storage Configuration
	Create Azure Arc data controller in direct connectivity mode using CLI
	Upload logs to Azure Monitor
	Create Azure Arc data controller using the CLI
	Upload metrics to Azure Monitor
	Create custom configuration templates
	Tutorial: Automated validation testing
	Quickstart: Deploy Azure Arc-enabled data services - indirectly connected mode - Azure CLI
	Upload usage data to Azure in indirect mode
az arcdata dc config	Storage Configuration
	Create Azure Arc data controller using the CLI
	Upload metrics to Azure Monitor
	Create custom configuration templates
	Tutorial: Automated validation testing
	Deploy the Azure Arc telemetry Router
az arcdata dc debug	Get logs to troubleshoot Azure Arc-enabled data services
az arcdata dc status	Upgrade a directly connected Azure Arc data controller using the CLI
	Upgrade an indirectly connected Azure Arc data controller using the CLI

az aro

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az aro	Tutorial: Create an Azure Red Hat OpenShift 4 cluster Create an Azure Red Hat OpenShift 4 private cluster Tutorial: Connect to an Azure Red Hat OpenShift 4 cluster Configure Microsoft Entra authentication for an Azure Red Hat OpenShift 4 cluster (Portal) Create an Azure Files StorageClass on Azure Red Hat OpenShift 4 Rotate service principal credentials for your Azure Red Hat OpenShift (ARO) Cluster Quickstart: Deploy an Azure Red Hat OpenShift cluster with an Azure Resource Manager template or Bicep file Create an Azure Red Hat OpenShift 4 cluster Application Backup Secure access to Azure Red Hat OpenShift with Azure Front Door Configure Microsoft Entra authentication for an Azure Red Hat OpenShift 4 cluster (CLI)

az artifacts

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az artifacts universal	Publish and download universal packages in Azure Artifacts

az attestation

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az attestation	Deploy an AKS cluster with Confidential Containers and a default policy Quickstart: Create an Azure Attestation provider by using Terraform Quickstart: Set up Azure Attestation with Azure CLI

Reference subgroup	Azure CLI article showing reference use
az attestation policy	Quickstart: Set up Azure Attestation with Azure CLI

az automation

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az automation hrwg	Migrate the existing agent-based hybrid workers to extension-based hybrid workers
	Deploy an extension-based Windows or Linux User Hybrid Runbook Worker in Azure Automation
az automation hrwg hrw	Migrate the existing agent-based hybrid workers to extension-based hybrid workers
	Deploy an extension-based Windows or Linux User Hybrid Runbook Worker in Azure Automation

az azurestackhci

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az azurestackhci image	Create Azure Stack HCI VM image using Azure Marketplace images
	Create Azure Stack HCI VM image using images in a local share
	Create Azure Stack HCI VM image using image in Azure Storage account

az backup

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az backup container	Quickstart: Back up SAP HANA System Replication on Azure VMs using Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Back up SQL databases in Azure VM using Azure CLI
	Back up Azure file shares with Azure CLI
	Restore Azure file shares with the Azure CLI
	Manage Azure file share backups with the Azure CLI
	Tutorial: Back up SAP HANA databases in an Azure VM using Azure CLI
az backup item	Selective disk backup and restore for Azure virtual machines
	Back up Azure file shares with Azure CLI
	Restore Azure file shares with the Azure CLI
	Manage Azure file share backups with the Azure CLI
	Manage SQL databases in an Azure VM using Azure CLI
	Tutorial: Back up SAP HANA databases in an Azure VM using Azure CLI
az backup job	Selective disk backup and restore for Azure virtual machines
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	Back up a virtual machine in Azure with the Azure CLI
	Restore a VM with Azure CLI
	Restore SQL databases in an Azure VM using Azure CLI
	Back up SQL databases in Azure VM using Azure CLI
	Back up Azure file shares with Azure CLI
	Restore Azure file shares with the Azure CLI
	Manage Azure file share backups with the Azure CLI
	Manage SQL databases in an Azure VM using Azure CLI
az backup policy	Back up an Azure VM using Enhanced policy
	Update the existing VM backup policy using CLI
	Back up SQL databases in Azure VM using Azure CLI
	Back up Azure file shares with Azure CLI

Reference subgroup	Azure CLI article showing reference use Manage Azure file share backups with the Azure CLI
	Manage SQL databases in an Azure VM using Azure CLI
	Tutorial: Back up SAP HANA databases in an Azure VM using Azure CLI
	Tutorial: Manage SAP HANA databases in an Azure VM using Azure CLI
<code>az backup protectable-item</code>	Quickstart: Back up SAP HANA System Replication on Azure VMs using Azure CLI
	Back up SQL databases in Azure VM using Azure CLI
	Manage SQL databases in an Azure VM using Azure CLI
	Tutorial: Back up SAP HANA databases in an Azure VM using Azure CLI
	Tutorial: Manage SAP HANA databases in an Azure VM using Azure CLI
<code>az backup protection</code>	Back up an Azure VM using Enhanced policy
	Delete an Azure Backup Recovery Services vault
	Selective disk backup and restore for Azure virtual machines
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	VM Snapshot Windows extension for Azure Backup
	Back up a virtual machine in Azure with the Azure CLI
	VM Snapshot Linux extension for Azure Backup
	Quickstart: Back up SAP HANA System Replication on Azure VMs using Azure CLI
	Back up SQL databases in Azure VM using Azure CLI
	Back up Azure file shares with Azure CLI
<code>az backup recoveryconfig</code>	Use Archive tier support
	Restore SAP HANA databases on Azure VMs
	Restore SQL databases in an Azure VM using Azure CLI
	Quickstart: Restore SAP HANA System Replication on Azure

Reference subgroup	Azure CLI article showing reference use
	VMs using Azure CLI
	Tutorial: Restore SAP HANA databases in an Azure VM using Azure CLI
az backup recoverypoint	Use Archive tier support Selective disk backup and restore for Azure virtual machines Encrypt backup data by using customer-managed keys
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	Restore files to a virtual machine in Azure
	Restore a VM with Azure CLI
	Restore SQL databases in an Azure VM using Azure CLI
	Restore Azure file shares with the Azure CLI
	Quickstart: Restore SAP HANA System Replication on Azure VMs using Azure CLI
	Tutorial: Restore SAP HANA databases in an Azure VM using Azure CLI
az backup restore	Use Archive tier support Selective disk backup and restore for Azure virtual machines Encrypt backup data by using customer-managed keys
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	Restore a VM with Azure CLI
	Restore SQL databases in an Azure VM using Azure CLI
	Restore Azure file shares with the Azure CLI
	Quickstart: Restore SAP HANA System Replication on Azure VMs using Azure CLI
	Tutorial: Restore SAP HANA databases in an Azure VM using Azure CLI
az backup restore files	Restore files to a virtual machine in Azure

Reference subgroup	Azure CLI article showing reference use
az backup vault	Delete an Azure Backup Recovery Services vault
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	Back up a virtual machine in Azure with the Azure CLI
	Restore SAP HANA databases on Azure VMs
	Quickstart: Create a Recovery Services vault using Bicep
	Quickstart: Create a Recovery Services vault using an ARM template
	Quickstart: Back up SAP HANA System Replication on Azure VMs using Azure CLI
	Restore SQL databases in an Azure VM using Azure CLI
	Back up SQL databases in Azure VM using Azure CLI
	Back up Azure file shares with Azure CLI
az backup vault backup-properties	Switch to Azure Monitor based alerts for Azure Backup
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	Back up a virtual machine in Azure with the Azure CLI
	Quickstart: Create a Recovery Services vault using Bicep
	Quickstart: Create a Recovery Services vault using an ARM template
	Quickstart: Back up SAP HANA System Replication on Azure VMs using Azure CLI
	Back up SQL databases in Azure VM using Azure CLI
	Tutorial: Back up SAP HANA databases in an Azure VM using Azure CLI
az backup vault encryption	Encrypt backup data by using customer-managed keys
az backup vault identity	Save and manage MARS agent passphrase securely in Azure Key Vault
	Encrypt backup data by using customer-managed keys

Reference subgroup	Azure CLI article showing reference use
az backup vault resource-guard-mapping	Configure Multi-user authorization using Resource Guard in Azure Backup

az baremetalinstance

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az baremetalinstance	Connect BareMetal Infrastructure instances in Azure
	Work with Azure Large Instances in the Azure portal

az batch

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az batch account	Quickstart: Use the Azure CLI to create a Batch account and run a job
	CLI example: Run a job and tasks with Azure Batch
	Use Azure Pipelines to build and deploy an HPC solution
	Quickstart: Create an Azure Batch account using Terraform
	Configure customer-managed keys for your Azure Batch account with Azure Key Vault and Managed Identity
	Manage Batch resources with Azure CLI
	CLI example: Add an application to an Azure Batch account
	CLI example: Create and manage a Linux pool in Azure Batch
	CLI example: Create a Batch account in Batch service mode
	CLI example: Create a Batch account in user subscription mode
az batch account keys	CLI example: Create a Batch account in Batch service mode
az batch application	Use Azure Pipelines to build and deploy an HPC solution
	CLI example: Add an application to an Azure Batch account

Reference subgroup	Azure CLI article showing reference use
az batch application package	Use Azure Pipelines to build and deploy an HPC solution CLI example: Add an application to an Azure Batch account
az batch file	Use Azure Batch CLI templates and file transfer
az batch job	Quickstart: Use the Azure CLI to create a Batch account and run a job CLI example: Run a job and tasks with Azure Batch
az batch location	Use Azure Batch CLI templates and file transfer Choose a VM size and image for compute nodes in an Azure Batch pool
az batch node	CLI example: Create and manage a Linux pool in Azure Batch
az batch pool	Quickstart: Use the Azure CLI to create a Batch account and run a job CLI example: Run a job and tasks with Azure Batch
	Use the Azure Compute Gallery to create a custom image pool
	Use Azure Pipelines to build and deploy an HPC solution
	Manage Batch resources with Azure CLI
	Use Azure Batch CLI templates and file transfer
	CLI example: Create and manage a Linux pool in Azure Batch
	Create a pool with disk encryption enabled
	CLI example: Create and manage a Windows pool in Azure Batch
az batch pool autoscale	CLI example: Create and manage a Windows pool in Azure Batch
az batch pool supported-images	Choose a VM size and image for compute nodes in an Azure Batch pool Provision Linux compute nodes in Batch pools
	CLI example: Create and manage a Linux pool in Azure Batch
az batch task	Quickstart: Use the Azure CLI to create a Batch account and run a job CLI example: Run a job and tasks with Azure Batch
	Manage Batch resources with Azure CLI

Reference subgroup	Azure CLI article showing reference use
az batch task file	Quickstart: Use the Azure CLI to create a Batch account and run a job

az bicep

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az bicep	Install Bicep tools Decompiling ARM template JSON to Bicep
	Bicep CLI commands
	Create private registry for Bicep modules
	Tutorial: Use a PostgreSQL service for development
	Quickstart: Publish Bicep modules to private module registry
	Tutorial: Create and use an Apache Kafka service for development
	Quickstart: Use Bicep to create and publish an Azure Managed Application definition
	Create Azure Managed Application that deploys storage account encrypted with customer-managed key

az billing

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az billing account	Programmatically create Azure Enterprise Agreement subscriptions with the latest APIs
	Programmatically create Azure subscriptions for a Microsoft Customer Agreement with the latest APIs
	Programmatically create Azure subscriptions for a Microsoft Partner Agreement with the latest APIs

Reference subgroup	Azure CLI article showing reference use
az billing customer	Programmatically create Azure subscriptions for a Microsoft Partner Agreement with the latest APIs
az billing enrollment-account	Grant access to create Azure Enterprise subscriptions (legacy)
	Programmatically create Azure subscriptions with legacy APIs
az billing profile	Programmatically create Azure subscriptions for a Microsoft Customer Agreement with the latest APIs

az blueprint

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az blueprint	Quickstart: Define and assign an Azure blueprint with the Azure CLI
az blueprint artifact policy	Quickstart: Define and assign an Azure blueprint with the Azure CLI
az blueprint artifact role	Quickstart: Define and assign an Azure blueprint with the Azure CLI
az blueprint artifact template	Quickstart: Define and assign an Azure blueprint with the Azure CLI
az blueprint assignment	Quickstart: Define and assign an Azure blueprint with the Azure CLI
az blueprint resource-group	Quickstart: Define and assign an Azure blueprint with the Azure CLI

az boards

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az boards	View, run, or email a work item query
az boards area project	Define area paths and assign to a team

Reference subgroup	Azure CLI article showing reference use
	Define iteration paths (sprints) and configure team iterations
az boards area team	Define area paths and assign to a team
	Define iteration paths (sprints) and configure team iterations
az boards iteration project	Define iteration paths (sprints) and configure team iterations
az boards iteration team	Define iteration paths (sprints) and configure team iterations
az boards work-item	Customize your pipeline
	Remove, delete, or restore work items in Azure Boards
	Add and update a work item
	View and add work items using the Work Items page
	Move work items from one team to another team
az boards work-item relation	Link work items to objects
	Reference guide for link types

az bot

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az bot	Provision and publish a bot
	Use Azure CLI to create or update an Azure Bot resource
az bot authsetting	Identity providers
az bot directline	Connect your bot to channels with Azure CLI
az bot email	Connect your bot to channels with Azure CLI
az bot facebook	Connect your bot to channels with Azure CLI
az bot msteams	Connect your bot to channels with Azure CLI
az bot skype	Connect your bot to channels with Azure CLI
az bot slack	Connect your bot to channels with Azure CLI
az bot sms	Connect your bot to channels with Azure CLI

Reference subgroup	Azure CLI article showing reference use
az bot telegram	Connect your bot to channels with Azure CLI

az capacity

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az capacity reservation	Create a Capacity Reservation
	Modify a Capacity Reservation
	Associate a VM to a Capacity Reservation group
	Remove a Virtual Machine Scale Set association from a Capacity Reservation group
	Remove a VM association from a Capacity Reservation group
az capacity reservation group	Manage node pools for a cluster in Azure Kubernetes Service (AKS)
	Create a Capacity Reservation
	Modify a Capacity Reservation
	Associate a Virtual Machine Scale Set with uniform orchestration to a Capacity Reservation group

az cdn

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az cdn custom-domain	Quickstart: Create an Azure CDN profile and endpoint using Terraform
	Create an Azure CDN profile and endpoint using the Azure CLI
az cdn endpoint	Use GitHub Actions workflow to deploy your static website in Azure Storage
	Quickstart: Create an Azure CDN profile and endpoint using Terraform

Reference subgroup	Azure CLI article showing reference use
	Create an Azure CDN profile and endpoint using the Azure CLI
az cdn origin	Create an Azure CDN profile and endpoint using the Azure CLI
az cdn origin-group	Create an Azure CDN profile and endpoint using the Azure CLI
az cdn profile	Create an Azure CDN profile and endpoint using the Azure CLI

az cloud

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az cloud	Azure CLI configuration
	Azure cloud management with the Azure CLI
	Use the Azure REST API with Azure CLI
	Use Microsoft Entra ID for authentication with Azure Database for PostgreSQL - Flexible Server
	Configuring Azure Files network endpoints
	Use Microsoft Entra ID for authentication with PostgreSQL
	Compare Azure Government and global Azure
	Azure NetApp Files for Azure Government
	Set up Microsoft Entra authentication for Azure Database for MySQL - Flexible Server
	Use GitHub Actions to connect to Azure

az cognitiveservices

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az cognitiveservices account	Configure Azure AI services virtual networks

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	JavaScript Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	Cognitive Services Anomaly Detector client library for Python
	Use Speech service through a private endpoint
	Quickstart: Create an Azure AI services resource using Terraform
	Create an Azure AI services resource using an ARM template
	Microsoft Entra authentication with the Speech SDK
	Configure data loss prevention for Azure AI services
	Tutorial: Isolate back-end communication in Azure App Service with Virtual Network integration
az cognitiveservices account deployment	Manage Azure OpenAI Service quota
	What is provisioned throughput?
	Get started using Provisioned Deployments on the Azure OpenAI Service
	Secure access to Azure OpenAI from Azure Kubernetes Service (AKS)
	Quickstart: Integrate Azure OpenAI
az cognitiveservices account keys	Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	JavaScript Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	Cognitive Services Anomaly Detector client library for Python
	Express.js app converts text to speech with Cognitive Services Speech
	Tutorial: Secure Cognitive Service connection from .NET App Service using Key Vault
	Build and deploy a Static Web app to Azure

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Secure Cognitive Service connection from PHP App Service using Key Vault
	Create a Personalizer resource
	Quickstart: Integrate Azure OpenAI
	Tutorial: Secure Cognitive Service connection from JavaScript App Service using Key Vault
az cognitiveservices account network-rule	Configure Azure AI services virtual networks
az cognitiveservices usage	Manage Azure OpenAI Service quota

az communication

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az communication	Quickstart: Create and manage Communication Services resources
	Get started with the chat hero sample
az communication identity	How to use Managed Identity with Azure Communication Services
az communication identity token	Quickstart: Join a room call
	Pre-Call diagnostic
	Adding visual effects to a video call

az confcom

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az confcom	Deploy an AKS cluster with Confidential Containers and a default policy

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Create an ARM template for a confidential container deployment with custom confidential computing enforcement policy

az confidentialledger

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az confidentialledger	Quickstart: Create a confidential ledger using the Azure CLI
az confidentialledger managedccfs	Quickstart: Create an Azure Managed CCF resource using Azure CLI

az config

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az config param-persist	Quickstart: Connect and query with Azure CLI with Azure Database for PostgreSQL - Flexible Server
	Quickstart: Connect with Azure Database for MySQL - Flexible Server by using Azure CLI
	Tutorial: Using Service Connector to build a Django app with Postgres on Azure App Service
az config	Tips for using the Azure CLI successfully
	How to update the Azure CLI
	Use and manage extensions with the Azure CLI
	Azure CLI configuration
	Output formats for Azure CLI commands
	How to use variables in Azure CLI commands
	How to manage Azure resource groups with the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Prepare your environment for the Azure CLI
	Sign in with Web Account Manager (WAM) using Azure CLI
	Azure CLI interactive mode

az configure

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Azure CLI configuration
	Quickstart: Deploy your first application to Azure Spring Apps
	Deploy and score a machine learning model by using an online endpoint
	Deploy Azure File Sync
	Troubleshooting online endpoints deployment and scoring
	Manage Azure Machine Learning environments with the CLI & SDK (v2)
	Work with models in Azure Machine Learning
	Set up AutoML training for tabular data with the Azure Machine Learning CLI and Python SDK
	Deploy models for scoring in batch endpoints
	Train models with Azure Machine Learning CLI, SDK, and REST API

az confluent

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az confluent organization	Manage the Confluent Cloud resource

Reference subgroup	Azure CLI article showing reference use
	QuickStart: Get started with Apache Kafka & Apache Flink on Confluent Cloud - Azure CLI

az connectedk8s

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az connectedk8s	Quickstart: Connect an existing Kubernetes cluster to Azure Arc
	Quickstart: Deploy Azure IoT Operations to an Arc-enabled Kubernetes cluster
	Use cluster connect to securely connect to Azure Arc-enabled Kubernetes clusters
	Create and manage custom locations on Azure Arc-enabled Kubernetes
	Prepare your Azure Arc-enabled Kubernetes cluster
	Create Kubernetes clusters using Azure CLI
	Use Azure RBAC on Azure Arc-enabled Kubernetes clusters (preview)
	Set up an Azure Arc-enabled Kubernetes cluster to run App Service, Functions, and Logic Apps (Preview)
	Manage asset configurations remotely
	Control access using Microsoft Entra ID and Kubernetes RBAC in AKS enabled by Azure Arc
az edk8s	Quickstart: Connect an existing Kubernetes cluster to Azure Arc
	Quickstart: Deploy Azure IoT Operations to an Arc-enabled Kubernetes cluster
	Prepare your Azure Arc-enabled Kubernetes cluster
	Set up an Azure Arc-enabled Kubernetes cluster to run App Service, Functions, and Logic Apps (Preview)
	Private connectivity for Arc-enabled Kubernetes clusters using private link (preview)
	Prerequisites to deploy the data controller in direct connectivity mode

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Enable Azure Container Apps on Azure Arc-enabled Kubernetes (Preview)
	Upgrade Azure Arc-enabled Kubernetes agents
	Diagnose connection issues for Azure Arc-enabled Kubernetes clusters
	Quickstart: Deploy Azure Arc-enabled data services - directly connected mode - Azure portal

az connectedmachine

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az connectedmachine	Manage Azure Monitor Agent
	Install and upgrade Azure Monitor Agent on Azure Arc-enabled servers
	Enable Azure VM extensions using the Azure CLI
	Troubleshoot Azure Arc VM management
	Monitor Nexus Kubernetes cluster
az connectedmachine extension	Manage Azure Monitor Agent
	What are Extended Security Updates for SQL Server?
	Overview of change tracking and inventory using Azure Monitoring Agent
	Configure SQL Server enabled by Azure Arc
	Install and upgrade Azure Monitor Agent on Azure Arc-enabled servers
	Automatic extension upgrade for Azure Arc-enabled servers
	SSH access to Azure Arc-enabled servers
	Enable Azure VM extensions using the Azure CLI
	Tutorial: Using automation to set up the Microsoft Entra admin for SQL Server

Reference subgroup	Azure CLI article showing reference use
	Disconnect your SQL Server instances from Azure Arc
az connectedmachine run-command	Remotely and securely configure servers using Run command (Preview)

az connectedvmware

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az connectedvmware vm	Deploy Arc-enabled VMware vSphere for Azure VMware Solution private cloud
	Enable virtual hardware and VM CRUD capabilities in a machine with Arc agent installed
az edvmware vcenter	Perform ongoing administration for Arc-enabled VMware vSphere

az connection

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az connection create	Use Spring Data JDBC with Azure SQL Database
	Use Spring Data JPA with Azure SQL Database
	Use Spring Data JDBC with Azure Database for PostgreSQL
	Use Spring Data JPA with Azure Database for MySQL
	Use Spring Data JPA with Azure Database for PostgreSQL
	Use Spring Data JDBC with Azure Database for MySQL

az consumption

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az consumption budget	Quickstart: Create a budget with an ARM template
	Quickstart: Create a budget with Bicep

az container

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az container	Quickstart: Deploy a container instance in Azure using the Azure CLI
	Deploy container instances into an Azure virtual network
	Mount an Azure file share in Azure Container Instances
	Set environment variables in container instances
	Tutorial: Build and deploy container images in the cloud with Azure Container Registry Tasks
	Troubleshoot common issues in Azure Container Instances
	Configure a GitHub Action to create a container instance
	Retrieve container logs and events in Azure Container Instances
	Tutorial: Deploy a container application to Azure Container Instances
	Tutorial: Deploy a multi-container group using a YAML file
az container app	Configure a GitHub Action to create a container instance

az containerapp

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az containerapp	Set scaling rules in Azure Container Apps
	Quickstart: Deploy your first container app with containerapp up
	Update and deploy changes in Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Use storage mounts in Azure Container Apps
	Manage secrets in Azure Container Apps
	Manage revisions in Azure Container Apps
	Configure Ingress for your app in Azure Container Apps
	Deploy Azure Container Apps with the az containerapp up command
	Managed identities in Azure Container Apps
	Deploy to Azure Container Apps from Azure Pipelines
az containerapp connected-env	Tutorial: Enable Azure Container Apps on Azure Arc-enabled Kubernetes (Preview)
	Tutorial: Create an Azure Container App on Azure Arc-enabled Kubernetes (Preview)
az containerapp connection	Service Connector internals
	Connect a container app to a cloud service with Service Connector
	Quickstart: Create a service connection in Azure Container Apps with the Azure CLI
	Get connection configurations added by Service Connector
az containerapp connection create	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Connect a container app to a cloud service with Service Connector
	Migrate a .NET application to use passwordless connections with Azure SQL Database
	Tutorial: Connect to PostgreSQL Database from a Java Quarkus Container App without secrets using a managed identity
	Migrate a Java application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Service Bus

Reference subgroup	Azure CLI article showing reference use
	Migrate an application to use passwordless connections with Azure Event Hubs
	Migrate an application to use passwordless connections with Azure Event Hubs for Kafka
	Migrate an application to use passwordless connections with Azure Database for MySQL
az containerapp dapr	Deploy Azure Container Apps with the az containerapp up command
	Enable Dapr on your container app
az containerapp env	Networking in Azure Container Apps environment
	Tutorial: Deploy self-hosted CI/CD runners and agents with Azure Container Apps jobs
	Create your first containerized functions on Azure Container Apps
	Provide a virtual network to an external Azure Container Apps environment
	Deploy Azure Container Apps with the az containerapp up command
	Tutorial: Build and deploy your app to Azure Container Apps
	Provide a virtual network to an internal Azure Container Apps environment
	Reliability in Azure Container Apps
	Blue-Green Deployment in Azure Container Apps
	Custom domain names and free managed certificates in Azure Container Apps (preview)
az containerapp env dapr-component	Tutorial: Deploy a Dapr application to Azure Container Apps using the Azure CLI
	Dapr components in Azure Container Apps
az containerapp env dapr-component resiliency	Dapr component resiliency (preview)
az containerapp env logs	View log streams in Azure Container Apps
az containerapp env storage	Use storage mounts in Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Create an Azure Files volume mount in Azure Container Apps
	How to enable your own persistent storage in Azure Spring Apps with the Standard consumption and dedicated plan
az containerapp env workload-profile	Manage workload profiles with the Azure CLI
	Tutorial: Connect to a Qdrant vector database in Azure Container Apps (preview)
	Quickstart: Provision an Azure Spring Apps Standard consumption and dedicated plan service instance
	Quickstart: Create an Azure Spring Apps Standard consumption and dedicated plan instance in an Azure Container Apps environment with a virtual network
az containerapp github-action	Set up GitHub Actions with Azure CLI in Azure Container Apps
	Configure continuous deployment for a Python web app in Azure Container Apps
az containerapp hostname	Custom domain names and free managed certificates in Azure Container Apps (preview)
az containerapp identity	Managed identities in Azure Container Apps
	Deploy to Azure Container Apps from Azure Pipelines
	Deploy to Azure Container Apps with GitHub Actions
	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate a Python application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Migrate an application to use passwordless connections with Azure Cosmos DB for NoSQL
	Migrate a .NET application to use passwordless connections with Azure SQL Database

Reference subgroup	Azure CLI article showing reference use
	Migrate a Java application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Service Bus
az containerapp ingress	Configure Ingress for your app in Azure Container Apps
	Deploy Azure Container Apps with the az containerapp up command
	Custom domain names and free managed certificates in Azure Container Apps (preview)
az containerapp ingress access-restriction	Set up IP ingress restrictions in Azure Container Apps
az containerapp ingress cors	Configure cross-origin resource sharing (CORS) for Azure Container Apps
	Tutorial: Connect to a Qdrant vector database in Azure Container Apps (preview)
az containerapp ingress traffic	Blue-Green Deployment in Azure Container Apps
	Traffic splitting in Azure Container Apps
az containerapp job	Jobs in Azure Container Apps
	Tutorial: Deploy self-hosted CI/CD runners and agents with Azure Container Apps jobs
	Managed identities in Azure Container Apps
	Azure Container Apps ARM and YAML template specifications
	Tutorial: Deploy an event-driven job with Azure Container Apps
	Create a job with Azure Container Apps
az containerapp job execution	Jobs in Azure Container Apps
	Tutorial: Deploy self-hosted CI/CD runners and agents with Azure Container Apps jobs
	Tutorial: Deploy an event-driven job with Azure Container Apps
	Create a job with Azure Container Apps
az containerapp logs	View log streams in Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Scale a container app
	Build and deploy a Python web app with Azure Container Apps and PostgreSQL
	Tutorial: Use a PostgreSQL service for development
	Tutorial: Connect to a Qdrant vector database in Azure Container Apps (preview)
	Configure continuous deployment for a Python web app in Azure Container Apps
	Tutorial: Create and use an Apache Kafka service for development
az containerapp registry	Deploy to Azure Container Apps from Azure Pipelines
	Deploy to Azure Container Apps with GitHub Actions
	Azure Container Apps image pull with managed identity
az containerapp replica	View log streams in Azure Container Apps
	Connect to a container console in Azure Container Apps
az containerapp resiliency	Service discovery resiliency (preview)
az containerapp revision	Manage revisions in Azure Container Apps
	View log streams in Azure Container Apps
	Connect to a container console in Azure Container Apps
	Azure Container Apps ARM and YAML template specifications
az containerapp revision label	Manage revisions in Azure Container Apps
	Blue-Green Deployment in Azure Container Apps
az containerapp secret	Deploy Azure Container Apps with the az containerapp up command
az containerapp service kafka	Tutorial: Create and use an Apache Kafka service for development
az containerapp service postgres	Tutorial: Use a PostgreSQL service for development
az containerapp service qdrant	Tutorial: Connect to a Qdrant vector database in Azure Container Apps (preview)

Reference subgroup	Azure CLI article showing reference use
az containerapp service redis	Connect to services in Azure Container Apps (preview)
Tutorial: Connect services in Azure Container Apps (preview)	

az cosmosdb

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az cosmosdb	Azure role-based access control in Azure Cosmos DB
	Quickstart: Azure Cosmos DB for Table for .NET
	Configure IP firewall in Azure Cosmos DB
	Azure Cosmos DB free tier
	Get started with Azure Cosmos DB for NoSQL using .NET
	Use system-assigned managed identities to access Azure Cosmos DB data
	What is Azure Cosmos DB analytical store?
	Monitor Azure Cosmos DB data by using diagnostic settings in Azure
	Delete items by partition key value - API for NoSQL (preview)
	JavaScript Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
az cosmosdb cassandra keyspace	Create an Azure Cosmos DB Cassandra API account, keyspace and table using Azure CLI
	Create an Azure Cosmos DB Cassandra API serverless account, keyspace and table using Azure CLI
	Use Azure CLI to create a API for Cassandra account, keyspace, and table with autoscale
	Throughput (RU/s) operations with Azure CLI for a keyspace or table for Azure Cosmos DB - API for Cassandra
az cosmosdb cassandra keyspace throughput	Throughput (RU/s) operations with Azure CLI for a keyspace or table for Azure Cosmos DB - API for Cassandra

Reference subgroup	Azure CLI article showing reference use
az cosmosdb cassandra table	Create an Azure Cosmos DB Cassandra API account, keyspace and table using Azure CLI
	Create an Azure Cosmos DB Cassandra API serverless account, keyspace and table using Azure CLI
	Use Azure CLI to create a API for Cassandra account, keyspace, and table with autoscale
	Throughput (RU/s) operations with Azure CLI for a keyspace or table for Azure Cosmos DB - API for Cassandra
az cosmosdb cassandra table throughput	Throughput (RU/s) operations with Azure CLI for a keyspace or table for Azure Cosmos DB - API for Cassandra
az cosmosdb copy	Create and manage container copy jobs in Azure Cosmos DB (Preview)
az cosmosdb database	Create an Azure Cosmos DB for NoSQL account, database and container using Azure CLI
az cosmosdb gremlin	Get the latest restorable timestamp for continuous backup accounts
az cosmosdb gremlin database	Quickstart: Azure Cosmos DB for Apache Gremlin library for .NET
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Quickstart: Azure Cosmos DB for Apache Gremlin library for Python
	Quickstart: Traverse vertices and edges with the Gremlin console and Azure Cosmos DB for Apache Gremlin
	Quickstart: Azure Cosmos DB for Apache Gremlin library for Node.js
	Create an Azure Cosmos DB for Gremlin account, database and graph using Azure CLI
	Use Azure CLI to create a Gremlin serverless account, database, and graph
	Use Azure CLI to create a API for Gremlin account, database, and graph with autoscale
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB - API for Gremlin

Reference subgroup	Azure CLI article showing reference use
az cosmosdb gremlin database throughput	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB - API for Gremlin
az cosmosdb gremlin graph	Configure and use Azure Synapse Link for Azure Cosmos DB Quickstart: Azure Cosmos DB for Apache Gremlin library for .NET
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Quickstart: Azure Cosmos DB for Apache Gremlin library for Python
	Quickstart: Traverse vertices and edges with the Gremlin console and Azure Cosmos DB for Apache Gremlin
	Quickstart: Azure Cosmos DB for Apache Gremlin library for Node.js
	Create an Azure Cosmos DB for Gremlin account, database and graph using Azure CLI
	Use Azure CLI to create a Gremlin serverless account, database, and graph
	Use Azure CLI to create a API for Gremlin account, database, and graph with autoscale
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB - API for Gremlin
az cosmosdb gremlin graph throughput	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB - API for Gremlin
az cosmosdb gremlin restorable-database	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb gremlin restorable-graph	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb gremlin restorable-resource	Restore an Azure Cosmos DB account that uses continuous backup mode

Reference subgroup	Azure CLI article showing reference use
az cosmosdb identity	Create and manage container copy jobs in Azure Cosmos DB (Preview)
	Configure managed identities with Microsoft Entra ID for your Azure Cosmos DB account
	Configure customer-managed keys for your Azure Cosmos DB account with Azure Managed HSM Key Vault
	Access Azure Key Vault from Azure Cosmos DB using a managed identity
az cosmosdb keys	Migrate data to Azure Cosmos DB using the desktop data migration tool
	Get started with Azure Cosmos DB for NoSQL using .NET
	Get started with Azure Cosmos DB for NoSQL using Python
	Quickstart: Azure Cosmos DB for MongoDB for .NET with the MongoDB driver
	Quickstart: Azure Cosmos DB for MongoDB for Python with MongoDB driver
	Quickstart: Azure Cosmos DB for Apache Gremlin library for .NET
	Quickstart: Azure Cosmos DB for MongoDB driver for Node.js
	Quickstart: Build an API for Table app with Python SDK and Azure Cosmos DB
	Manage Azure Cosmos DB for NoSQL resources using Azure CLI
	Index and query GeoJSON location data in Azure Cosmos DB for NoSQL
az cosmosdb locations	Update periodic backup storage redundancy for Azure Cosmos DB
az cosmosdb mongocluster	Monitor Azure Cosmos DB for MongoDB vCore diagnostics logs with Azure Monitor
az cosmosdb mongodb	Get the latest restorable timestamp for continuous backup accounts
az cosmosdb mongodb collection	Merge partitions in Azure Cosmos DB (preview)
	Redistribute throughput across partitions (preview)

Reference subgroup	Azure CLI article showing reference use
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Build and run a containerized Python web app locally with MongoDB
	Create a database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Create a serverless database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB for MongoDB
	Create a database with autoscale and shared collections for API for MongoDB for Azure Cosmos DB using Azure CLI
az cosmosdb mongodb collection throughput	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB for MongoDB
az cosmosdb mongodb database	Merge partitions in Azure Cosmos DB (preview)
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Build and run a containerized Python web app locally with MongoDB
	Create a database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Create a serverless database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB for MongoDB
	Create a database with autoscale and shared collections for API for MongoDB for Azure Cosmos DB using Azure CLI
az cosmosdb mongodb database throughput	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB for MongoDB
az cosmosdb mongodb restorable-collection	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)

Reference subgroup	Azure CLI article showing reference use
az cosmosdb mongodb restorable-database	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb mongodb restorable-resource	Restore an Azure Cosmos DB account that uses continuous backup mode
az cosmosdb mongodb role definition	Configure role-based access control in Azure Cosmos DB for MongoDB
az cosmosdb mongodb user definition	Configure role-based access control in Azure Cosmos DB for MongoDB
az cosmosdb network-rule	Configure access to Azure Cosmos DB from virtual networks (VNet)
	Connect an existing Azure Cosmos DB account with virtual network service endpoints using Azure CLI
az cosmosdb restorable-database-account	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Manage permissions to restore an Azure Cosmos DB account
az cosmosdb service	Materialized views in Azure Cosmos DB for Apache Cassandra (preview)
az cosmosdb sql	Get the latest restorable timestamp for continuous backup accounts
az cosmosdb sql container	Migrate data to Azure Cosmos DB using the desktop data migration tool
	JavaScript Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	Merge partitions in Azure Cosmos DB (preview)
	Manage Azure Cosmos DB for NoSQL resources using Azure CLI
	Redistribute throughput across partitions (preview)
	Index and query GeoJSON location data in Azure Cosmos DB for NoSQL

Reference subgroup	Azure CLI article showing reference use
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
	Measure Azure Cosmos DB for NoSQL performance with a benchmarking framework
	Create an Azure Cosmos DB for NoSQL account, database and container using Azure CLI
<code>az cosmosdb sql container throughput</code>	Manage Azure Cosmos DB for NoSQL resources using Azure CLI
	Throughput (RU/s) operations with Azure CLI for a database or container for Azure Cosmos DB for NoSQL
<code>az cosmosdb sql database</code>	Migrate data to Azure Cosmos DB using the desktop data migration tool
	Use system-assigned managed identities to access Azure Cosmos DB data
	JavaScript Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	Merge partitions in Azure Cosmos DB (preview)
	Manage Azure Cosmos DB for NoSQL resources using Azure CLI
	Index and query GeoJSON location data in Azure Cosmos DB for NoSQL
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
	Measure Azure Cosmos DB for NoSQL performance with a benchmarking framework
	Create an Azure Cosmos DB for NoSQL account, database and container using Azure CLI
<code>az cosmosdb sql database throughput</code>	Manage Azure Cosmos DB for NoSQL resources using Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Throughput (RU/s) operations with Azure CLI for a database or container for Azure Cosmos DB for NoSQL
az cosmosdb sql restorable-container	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb sql restorable-database	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb sql restorable-resource	Restore an Azure Cosmos DB account that uses continuous backup mode
az cosmosdb sql role assignment	Configure role-based access control with Microsoft Entra ID for your Azure Cosmos DB account
	Use system-assigned managed identities to access Azure Cosmos DB data
	How to use managed identities to connect to Azure Cosmos DB from an Azure virtual machine
	Ingest data from Azure Cosmos DB into Azure Data Explorer
	Create and manage container copy jobs in Azure Cosmos DB (Preview)
	Migrate an application to use passwordless connections with Azure Cosmos DB for NoSQL
	Cosmos DB data source for a resolver
	Set up an indexer connection to Azure Cosmos DB via a managed identity
	Create and delete routes and endpoints by using the Azure CLI
az cosmosdb sql role definition	Configure role-based access control with Microsoft Entra ID for your Azure Cosmos DB account
	Use system-assigned managed identities to access Azure Cosmos DB data
	Migrate an application to use passwordless connections with Azure Cosmos DB for NoSQL

Reference subgroup	Azure CLI article showing reference use
	Cosmos DB data source for a resolver
az cosmosdb table	Quickstart: Build an API for Table app with Python SDK and Azure Cosmos DB
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
	Quickstart: Build a API for Table app with Node.js and Azure Cosmos DB
	Get the latest restorable timestamp for continuous backup accounts
	Quickstart: Build a API for Table app with Java SDK and Azure Cosmos DB
	Create an Azure Cosmos DB Table API account and table using Azure CLI
	Use Azure CLI to create an Azure Cosmos DB for Table serverless account and table
	Use Azure CLI to create an Azure Cosmos DB for Table account and table with autoscale
	Throughput (RU/s) operations with Azure CLI for a table for Azure Cosmos DB for Table
az cosmosdb table restorable-resource	Restore an Azure Cosmos DB account that uses continuous backup mode
az cosmosdb table restorable-table	Restore an Azure Cosmos DB account that uses continuous backup mode
	Restore a deleted container or database to the same Azure Cosmos DB account (preview)
az cosmosdb table throughput	Throughput (RU/s) operations with Azure CLI for a table for Azure Cosmos DB for Table

az costmanagement

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az costmanagement export	Tutorial: Create and manage exported data

Reference subgroup	Azure CLI article showing reference use View and download your Azure usage and charges
Get usage data with the Azure CLI	

az customlocation

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az customlocation	Uninstall Azure Arc-enabled data services
	Create and manage custom locations on Azure Arc-enabled Kubernetes
	Create Kubernetes clusters using Azure CLI
	Commission the AKS cluster
	Create your first function on Azure Arc (preview)
	Set up an Azure Arc-enabled Kubernetes cluster to run App Service, Functions, and Logic Apps (Preview)
	Create Azure Stack HCI VM image using Azure Marketplace images
	Create Azure Stack HCI VM image using images in a local share
	Create Azure Stack HCI VM image using image in Azure Storage account
	Tutorial: Enable Azure Container Apps on Azure Arc-enabled Kubernetes (Preview)

az custom-providers

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az custom-providers resource-provider	Quickstart: Create Azure Custom Resource Provider and deploy custom resources

az databox

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az databox job	Tutorial: Order Azure Data Box
	Quickstart: Deploy Azure Data Box Disk using the Azure portal
	Quickstart: Deploy Azure Data Box Heavy using the Azure portal

az databoxedge

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az databoxedge device	Tutorial: Prepare to deploy Azure Stack Edge Pro R
	Tutorial: Prepare to deploy Azure Stack Edge Mini R
az databoxedge order	Tutorial: Prepare to deploy Azure Stack Edge Pro R
	Tutorial: Prepare to deploy Azure Stack Edge Mini R

az databricks

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az databricks workspace	Get Microsoft Entra ID (formerly Azure Active Directory) tokens for service principals
	Azure CLI login with a Microsoft Entra ID service principal
	Get Microsoft Entra ID (formerly Azure Active Directory) tokens for users by using the Azure CLI
	Build a data pipeline by using Azure Data Factory, DevOps, and machine learning
	Enable customer-managed keys for managed services
	Azure CLI login with an Azure Databricks user account
	Configure customer-managed keys for Azure managed disks

Reference subgroup	Azure CLI article showing reference use
	Get Microsoft Entra ID (formerly Azure Active Directory) tokens for Microsoft Entra ID service principals by using the Azure CLI
	Configure double encryption for DBFS root
	Configure HSM customer-managed keys for Azure managed disks

az datafactory

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az datafactory	Build a data pipeline by using Azure Data Factory, DevOps, and machine learning
	Quickstart: Create an Azure Data Factory using Azure CLI
	Tutorial: Clean up resources
az datafactory dataset	Quickstart: Create an Azure Data Factory using Azure CLI
az datafactory integration-runtime	Tutorial: Clean up resources
az datafactory linked-service	Quickstart: Create an Azure Data Factory using Azure CLI
az datafactory managed-private-endpoint	Connect to an Azure Data Factory privately networked pipeline with Azure Database for PostgreSQL - Flexible Server by using Azure Private Link
az datafactory pipeline	Quickstart: Create an Azure Data Factory using Azure CLI
az datafactory pipeline-run	Quickstart: Create an Azure Data Factory using Azure CLI
az datafactory trigger	Create a trigger that runs a pipeline on a schedule
	Create a trigger that runs a pipeline on a tumbling window
az datafactory trigger-run	Create a trigger that runs a pipeline on a schedule
	Create a trigger that runs a pipeline on a tumbling window

az datamigration

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az datamigration sql-managed-instance	Migrate databases at scale using automation (Preview)
az datamigration sql-service	Migrate databases at scale using automation (Preview)

az dataprotection

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dataprotection backup-instance	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure Blobs in a storage account using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Restore Azure Kubernetes Service using Azure CLI
	Back up Azure Managed Disks using Azure CLI
	Restore Azure Blobs to point-in-time using Azure CLI
	Restore Azure PostgreSQL databases using Azure CLI
	Restore Azure Managed Disks using Azure CLI
az dataprotection backup-instance restore	Restore Azure Kubernetes Service using Azure CLI
	Restore Azure Blobs to point-in-time using Azure CLI
	Restore Azure PostgreSQL databases using Azure CLI
	Restore Azure Managed Disks using Azure CLI
az dataprotection backup-policy	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure Blobs in a storage account using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Back up Azure Managed Disks using Azure CLI

Reference subgroup	Azure CLI article showing reference use
az dataprotection backup-policy retention-rule	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Back up Azure Managed Disks using Azure CLI
az dataprotection backup-policy tag	Back up Azure PostgreSQL databases using Azure CLI
az dataprotection backup-policy trigger	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Back up Azure Managed Disks using Azure CLI
az dataprotection backup-vault	Manage Azure Kubernetes Service backups using Azure Backup
	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure Blobs in a storage account using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Back up Azure Managed Disks using Azure CLI
az dataprotection job	Back up Azure Kubernetes Service using Azure CLI
	Back up Azure PostgreSQL databases using Azure CLI
	Restore Azure Kubernetes Service using Azure CLI
	Back up Azure Managed Disks using Azure CLI
	Restore Azure Blobs to point-in-time using Azure CLI
	Restore Azure PostgreSQL databases using Azure CLI
	Restore Azure Managed Disks using Azure CLI
az dataprotection recovery-point	Restore Azure Kubernetes Service using Azure CLI
	Restore Azure PostgreSQL databases using Azure CLI
	Restore Azure Managed Disks using Azure CLI
az dataprotection resource-guard	Configure Multi-user authorization using Resource Guard in Azure Backup
az dataprotection restorable-time-	Restore Azure Blobs to point-in-time using Azure CLI

Reference subgroup	Azure CLI article showing reference use
range	

az datashare

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az datashare	Tutorial: Share data using Azure Data Share
az datashare account	Tutorial: Share data using Azure Data Share
	Tutorial: Accept and receive data using Azure Data Share
az datashare consumer-invitation	Tutorial: Accept and receive data using Azure Data Share
az datashare consumer-source-data-set	Tutorial: Accept and receive data using Azure Data Share
az datashare data-set-mapping	Tutorial: Accept and receive data using Azure Data Share
az datashare invitation	Tutorial: Share data using Azure Data Share
az datashare share-subscription	Tutorial: Accept and receive data using Azure Data Share
az datashare trigger	Tutorial: Accept and receive data using Azure Data Share

az dedicated-hsm

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az dedicated-hsm	Tutorial: Deploying HSMs into an existing virtual network using the Azure CLI
	Tutorial: Create a payment HSM
	Troubleshooting the Azure Dedicated HSM service

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create an Azure Dedicated HSM by using the Azure CLI
	Tutorial: View your payment HSMs
	Quickstart: Create an Azure Payment HSM with the Azure CLI
	Tutorial: Create a payment HSM with host and management port in different virtual networks using ARM template
	Tutorial: Remove a commissioned payment HSM
	Quickstart: Create an Azure payment HSM using an ARM template

az deployment

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az deployment group	Tips for using the Azure CLI successfully
	Create a storage account
	Manage Azure Monitor Agent
	View deployment history with Azure Resource Manager
	Manage Azure Resource Groups by using Azure CLI
	Tutorial: Create and deploy your first ARM template
	Provision and publish a bot
	Create parameters files for Bicep deployment
	How to deploy resources with Bicep and Azure CLI
	Configure an AKS cluster
az deployment mg	View deployment history with Azure Resource Manager
	How to deploy resources with Bicep and Azure CLI
	How to use Azure Resource Manager (ARM) deployment templates with Azure CLI
	Bicep deployment what-if operation

Reference subgroup	Azure CLI article showing reference use
	Programmatically create Azure Enterprise Agreement subscriptions with the latest APIs
	ARM template deployment what-if operation
	Find error codes
	Programmatically create Azure subscriptions for a Microsoft Customer Agreement with the latest APIs
	Use Azure CLI to export a template
	Enable debug logging
az deployment operation group	View deployment history with Azure Resource Manager
	Find error codes
	Enable debug logging
	Troubleshoot Azure Database for PostgreSQL - Flexible Server CLI errors
	Troubleshoot Azure Database for MySQL - Flexible Server CLI errors
az deployment operation mg	View deployment history with Azure Resource Manager
	Find error codes
	Enable debug logging
az deployment operation sub	View deployment history with Azure Resource Manager
	Find error codes
	Enable debug logging
az deployment operation tenant	View deployment history with Azure Resource Manager
	Find error codes
	Enable debug logging
az deployment sub	View deployment history with Azure Resource Manager
	How to deploy resources with Bicep and Azure CLI

Reference subgroup	Azure CLI article showing reference use
	How to use Azure Resource Manager (ARM) deployment templates with Azure CLI
	Onboard a customer to Azure Lighthouse
	Assign Azure roles using Azure Resource Manager templates
	Subscription deployments with ARM templates
	Bicep deployment what-if operation
	Create resource groups by using Bicep
	Subscription deployments with Bicep files
	ARM template deployment what-if operation
az deployment tenant	View deployment history with Azure Resource Manager
	How to deploy resources with Bicep and Azure CLI
	How to use Azure Resource Manager (ARM) deployment templates with Azure CLI
	Bicep deployment what-if operation
	ARM template deployment what-if operation
	Find error codes
	Use Azure CLI to export a template
	Tenant deployments with ARM templates
	Enable debug logging
	Tenant deployments with Bicep file

az deployment-scripts

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az deployment-scripts	Use deployment scripts in ARM templates

az desktopvirtualization

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az desktopvirtualization applicationgroup	Deploy Azure Virtual Desktop
	Use Azure CLI and Azure PowerShell with Azure Virtual Desktop
	Delete a host pool
az desktopvirtualization hostpool	Deploy Azure Virtual Desktop
	Add session hosts to a host pool
	Configure personal desktop assignment
	Set up Private Link with Azure Virtual Desktop
	Configure a host pool as a validation environment
	Use Azure CLI and Azure PowerShell with Azure Virtual Desktop
	Delete a host pool
az desktopvirtualization workspace	Deploy Azure Virtual Desktop
	Add session hosts to a host pool
	Set up Private Link with Azure Virtual Desktop
	Use Azure CLI and Azure PowerShell with Azure Virtual Desktop

az devcenter

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az devcenter admin catalog	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments
	Create and configure a dev center for Azure Deployment Environments by using the Azure CLI
az devcenter admin devbox-definition	Configure hibernation in Microsoft Dev Box

Reference subgroup	Azure CLI article showing reference use
az devcenter admin devcenter	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments Create and configure a project by using the Azure CLI
	Create and configure a dev center for Azure Deployment Environments by using the Azure CLI
az devcenter admin environment-type	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments Create and configure a dev center for Azure Deployment Environments by using the Azure CLI
az devcenter admin project	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments Create and configure a project by using the Azure CLI
az devcenter admin project-allowed-environment-type	Create and configure a project by using the Azure CLI
az devcenter admin project-environment-type	Tutorial: Deploy environments in CI/CD by using GitHub and Azure Deployment Environments Create and configure a project by using the Azure CLI
az devcenter admin schedule	Auto-stop your Dev Boxes on schedule
az devcenter dev dev-box	Hibernate a dev box in Microsoft Dev Box
az devcenter dev environment	Create and access an environment by using the Azure CLI Add and configure an environment definition
az devcenter dev environment-definition	Create and access an environment by using the Azure CLI
az devcenter dev environment-type	Create and access an environment by using the Azure CLI

az devops

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az devops	Define variables

Reference subgroup	Azure CLI article showing reference use
	Azure Pipelines agents
	Create your first pipeline
	Add & use variable groups
	Branch policies and settings
	Create pull requests
	Create and manage agent pools
	Use service principals & managed identities
	Add organization users and manage access
	Remove, delete, or restore work items in Azure Boards
<code>az devops admin banner</code>	Add and manage information banners in Azure DevOps
<code>az devops extension</code>	Install extensions
<code>az devops project</code>	Create a project in Azure DevOps
	Use a variable group's secret and nonsecret variables in an Azure Pipeline
	Delete a project
<code>az devops security group</code>	Add organization users and manage access
	Add and manage security groups
<code>az devops security group membership</code>	Add and manage security groups
<code>az devops security permission</code>	Manage permissions with command line tool
<code>az devops security permission namespace</code>	Manage permissions with command line tool
<code>az devops service-endpoint</code>	Use a variable group's secret and nonsecret variables in an Azure Pipeline
	Azure DevOps CLI service endpoint
<code>az devops service-endpoint azurerm</code>	Azure DevOps CLI service endpoint
<code>az devops service-endpoint github</code>	Use a variable group's secret and nonsecret variables in an Azure Pipeline

Reference subgroup	Azure CLI article showing reference use
	Azure DevOps CLI service endpoint
az devops team	Add users or groups to a team or project
	Create or add a team
	Rename or remove a team
az devops user	Add organization users and manage access
	Export a list of users and their access levels
	Remove users from Azure DevOps
az devops wiki	Add and edit wiki pages
	Create a wiki for your project
	Publish a Git repo to a wiki
	Manage wikis with the CLI
az devops wiki page	Add and edit wiki pages
	Create a wiki for your project
	Manage wikis with the CLI

az disk

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az disk	How to use variables in Azure CLI commands
	Expand virtual hard disks on a Linux VM
	Create and use a volume with Azure Disks in Azure Kubernetes Service (AKS)
	Download a Windows VHD from Azure
	Change the disk type of an Azure managed disk
	Deploy a Premium SSD v2

Reference subgroup	Azure CLI article showing reference use
	Migrate Virtual Machines and Virtual Machine Scale Sets to availability zone support
	Create an incremental snapshot for managed disks
	Using Azure ultra disks
	Enable shared disk

az disk-access

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az disk-access	Azure CLI - Restrict import/export access for managed disks with Private Links

az disk-encryption-set

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az disk-encryption-set	Bring your own keys (BYOK) with Azure managed disks in Azure Kubernetes Service (AKS)
	Use the Azure CLI to enable end-to-end encryption using encryption at host
	Copy managed disks to same or different subscription with CLI
	Encrypt managed disks with cross-tenant customer-managed keys
	Create a managed disk from a snapshot with CLI (Linux)
	Copy snapshot of a managed disk to same or different subscription with CLI
	Use the Azure CLI to enable server-side encryption with customer-managed keys for managed disks
	Quickstart: Deploy confidential VM with ARM template

Reference subgroup	Azure CLI article showing reference use
	Encrypt OS disks with a customer-managed key on Azure Red Hat OpenShift
	Quickstart: Create a confidential VM with the Azure CLI

az dla

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dla account	Manage Azure Data Lake Analytics using the Azure CLI
	Get started with Azure Data Lake Analytics using Azure CLI
az dla account blob-storage	Manage Azure Data Lake Analytics using the Azure CLI
az dla account data-lake-store	Manage Azure Data Lake Analytics using the Azure CLI
az dla job	Manage Azure Data Lake Analytics using the Azure CLI
	Get started with Azure Data Lake Analytics using Azure CLI
az dla job pipeline	Manage Azure Data Lake Analytics using the Azure CLI
az dla job recurrence	Manage Azure Data Lake Analytics using the Azure CLI

az dls

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dls account	Get started with Azure Data Lake Storage Gen1 using the Azure CLI
	Get started with Azure Data Lake Analytics using Azure CLI
az dls fs	Get started with Azure Data Lake Storage Gen1 using the Azure CLI
	Get started with Azure Data Lake Analytics using Azure CLI
az dls fs access	Get started with Azure Data Lake Storage Gen1 using the Azure CLI

az dms

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dms	Tutorial: Migrate PostgreSQL to Azure Database for PostgreSQL online using DMS (classic) via the Azure CLI
az dms project	Tutorial: Migrate PostgreSQL to Azure Database for PostgreSQL online using DMS (classic) via the Azure CLI
az dms project task	Tutorial: Migrate PostgreSQL to Azure Database for PostgreSQL online using DMS (classic) via the Azure CLI

az dns-resolver

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dns-resolver	Quickstart: Create an Azure DNS Private Resolver using Bicep
az dns-resolver forwarding-ruleset	Quickstart: Create an Azure DNS Private Resolver using Bicep
az dns-resolver inbound-endpoint	Quickstart: Create an Azure DNS Private Resolver using Bicep
az dns-resolver outbound-endpoint	Quickstart: Create an Azure DNS Private Resolver using Bicep
az dns-resolver vnet-link	Quickstart: Create an Azure DNS Private Resolver using Bicep

az dt

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az dt	Ingest IoT Hub telemetry into Azure Digital Twins Integrate Azure Digital Twins with Azure Time Series Insights

Reference subgroup	Azure CLI article showing reference use Create endpoints in Azure Digital Twins
	Set up twin-to-twin event handling
	Set up an Azure Digital Twins instance and authentication (CLI)
	Tutorial: Create an Azure Digital Twins graph using the Azure CLI
	Enable private access to Azure Digital Twins using Private Link
<code>az dt data-history connection create</code>	Create a data history connection for Azure Digital Twins
<code>az dt endpoint</code>	Tutorial: Build out an end-to-end solution
	Set up twin-to-twin event handling
<code>az dt endpoint create</code>	Tutorial: Build out an end-to-end solution
	Integrate Azure Digital Twins data into an Azure Maps indoor map
	Integrate Azure Digital Twins with Azure Time Series Insights
	Create endpoints in Azure Digital Twins
	Set up twin-to-twin event handling
<code>az dt identity</code>	Set up an Azure Digital Twins instance and authentication (CLI)
<code>az dt job deletion</code>	Quickstart - Get started with 3D Scenes Studio (preview) for Azure Digital Twins
	Tutorial: Coding with the Azure Digital Twins SDK
	Tutorial: Create an Azure Digital Twins graph using the Azure CLI
<code>az dt model</code>	Quickstart - Get started with 3D Scenes Studio (preview) for Azure Digital Twins
	Ingest IoT Hub telemetry into Azure Digital Twins
	Tutorial: Coding with the Azure Digital Twins SDK
	Automanage devices in Azure Digital Twins using Device Provisioning Service (DPS)
	Tutorial: Create an Azure Digital Twins graph using the Azure CLI
<code>az dt role-assignment</code>	Tutorial: Build out an end-to-end solution
	Ingest IoT Hub telemetry into Azure Digital Twins

Reference subgroup	Azure CLI article showing reference use
	Troubleshoot Azure Digital Twins failed service request: Error 403 (Forbidden)
	Create an app registration to use with Azure Digital Twins
	Integrate Azure Digital Twins with Azure Time Series Insights
	Set up twin-to-twin event handling
	Set up an Azure Digital Twins instance and authentication (CLI)
	Troubleshoot Azure Digital Twins Explorer: Authentication errors
az dt route	Tutorial: Build out an end-to-end solution
	Integrate Azure Digital Twins data into an Azure Maps indoor map
	Integrate Azure Digital Twins with Azure Time Series Insights
	Set up twin-to-twin event handling
az dt twin	Quickstart - Get started with 3D Scenes Studio (preview) for Azure Digital Twins
	Ingest IoT Hub telemetry into Azure Digital Twins
	Tutorial: Coding with the Azure Digital Twins SDK
	Automanage devices in Azure Digital Twins using Device Provisioning Service (DPS)
	Integrate Azure Digital Twins with Azure Time Series Insights
	Azure Digital Twins CLI command set
	Tutorial: Create an Azure Digital Twins graph using the Azure CLI
az dt twin relationship	Quickstart - Get started with 3D Scenes Studio (preview) for Azure Digital Twins
	Tutorial: Coding with the Azure Digital Twins SDK
	Tutorial: Create an Azure Digital Twins graph using the Azure CLI

az elastic-san

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az elastic-san	Deploy an Elastic SAN
	Configure network access for Azure Elastic SAN
	Increase the size of an Elastic SAN
	Delete an Elastic SAN
az elastic-san volume	Deploy an Elastic SAN
	Snapshot Azure Elastic SAN volumes (preview)
	Connect Azure Elastic SAN volumes to an Azure Kubernetes Service cluster
	Increase the size of an Elastic SAN
az elastic-san volume-snapshot	Configure customer-managed keys for an Azure Elastic SAN
	Delete an Elastic SAN
	Snapshot Azure Elastic SAN volumes (preview)
	Manage customer-managed keys for Azure Elastic SAN
az elastic-san volume-group	Deploy an Elastic SAN
	Configure network access for Azure Elastic SAN
	Configure customer-managed keys for an Azure Elastic SAN
	Delete an Elastic SAN

az eventgrid

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az eventgrid domain	Manage topics and publish events using event domains
	Assign a managed identity to an Event Grid custom topic or domain
az eventgrid domain key	Manage topics and publish events using event domains
	Get access keys for Event Grid resources (topics or domains)

Reference subgroup	Azure CLI article showing reference use
az eventgrid event-subscription	Event Grid message delivery and retry Azure Service Bus to Event Grid integration overview
	Service Bus queues and topics as event handlers for Azure Event Grid events
	Use CloudEvents v1.0 schema with Event Grid
	Set dead-letter location and retry policy
	Quickstart: Route custom events to web endpoint with Azure CLI and Event Grid
	Quickstart: Route storage events to web endpoint with Azure CLI
	Tutorial: Build out an end-to-end solution
	Trigger applications, processes, or CI/CD workflows based on Azure Machine Learning events (preview)
	Ingest IoT Hub telemetry into Azure Digital Twins
az eventgrid namespace	Publish to namespace topics and consume events in Azure Event Grid
	Quickstart: Publish and subscribe to MQTT messages on an Event Grid namespace with the Azure CLI
	Publish and deliver events using namespace topics (preview)
	Tutorial: Route MQTT messages to Azure Event Hubs from Azure Event Grid with the Azure CLI
	Tutorial: Configure MQTT bridge between IoT MQ and Azure Event Grid
az eventgrid namespace client	Quickstart: Publish and subscribe to MQTT messages on an Event Grid namespace with the Azure CLI
az eventgrid namespace permission-binding	Quickstart: Publish and subscribe to MQTT messages on an Event Grid namespace with the Azure CLI
az eventgrid namespace topic	Publish to namespace topics and consume events in Azure Event Grid
	Publish and deliver events using namespace topics (preview)
az eventgrid namespace topic	Publish to namespace topics and consume events in Azure

Reference subgroup	Azure CLI article showing reference use
event-subscription	Event Grid
az eventgrid namespace topic-space	Quickstart: Publish and subscribe to MQTT messages on an Event Grid namespace with the Azure CLI
az eventgrid partner topic event-subscription	Tutorial: Configure MQTT bridge between IoT MQ and Azure Event Grid
az eventgrid system-topic	Service Bus queues and topics as event handlers for Azure Event Grid events
az eventgrid system-topic	Tutorial: Route policy state change events to Event Grid with Azure CLI
	MQTT Clients Life Cycle Events
az eventgrid system-topic	Create, view, and manage Event Grid system topics using Azure CLI
az eventgrid system-topic	Subscribe to Azure Resource Management events in Event Grid (Preview)
	Create a Managed Application to Store Blob Digests
az eventgrid system-topic	Subscribe to events raised by Azure Resource Notifications - Health Resources system topic (Preview)
az eventgrid system-topic event-subscription	Deliver events to Microsoft Entra protected endpoints
	Tutorial: Route policy state change events to Event Grid with Azure CLI
az eventgrid system-topic	MQTT Clients Life Cycle Events
az eventgrid system-topic	Create, view, and manage Event Grid system topics using Azure CLI
az eventgrid system-topic	Subscribe to Azure Resource Management events in Event Grid (Preview)
az eventgrid system-topic	Create a Managed Application to Store Blob Digests
az eventgrid system-topic	Subscribe to events raised by Azure Resource Notifications - Health Resources system topic (Preview)
az eventgrid topic	Route custom events to web endpoint with the Azure portal and Azure Event Grid
az eventgrid topic	Quickstart: Route custom events to an Azure Function with Event Grid

Reference subgroup	Azure CLI article showing reference use
	Use CloudEvents v1.0 schema with Event Grid
	Set dead-letter location and retry policy
	Quickstart: Route custom events to web endpoint with Azure CLI and Event Grid
	Configure private endpoints for Azure Event Grid custom topics or domains
	Publish events to Azure Event Grid custom topics using access keys
	Tutorial: Build out an end-to-end solution
	Filter events for Event Grid
	Quickstart: Route custom events to Azure Event Hubs with Azure CLI and Event Grid
<code>az eventgrid topic key</code>	Route custom events to web endpoint with the Azure portal and Azure Event Grid
	Quickstart: Route custom events to an Azure Function with Event Grid
	Quickstart: Route custom events to web endpoint with Azure CLI and Event Grid
	Publish events to Azure Event Grid custom topics using access keys
	Filter events for Event Grid
	Quickstart: Route custom events to Azure Event Hubs with Azure CLI and Event Grid
	Quickstart: Route custom events to Azure Queue storage via Event Grid using Azure CLI
	Durable Functions publishing to Azure Event Grid
	Create custom topic and subscribe to events for an Azure subscription with Azure CLI
	Get access keys for Event Grid resources (topics or domains)
<code>az eventgrid topic private-endpoint-connection</code>	Configure private endpoints for Azure Event Grid custom topics or domains

az eventhubs

 Expand table

Reference subgroup	Azure CLI article showing reference use
az eventhubs eventhub	Event delivery with a managed identity
	Dynamically add partitions to an event hub (Apache Kafka topic)
	Quickstart: Subscribe to Azure Kubernetes Service (AKS) events with Azure Event Grid
	Quickstart: Route custom events to Azure Event Hubs with Azure CLI and Event Grid
	Grant managed identity the access to Event Grid destination
	Quickstart: Create an event hub using Azure CLI
	Publish and deliver events using namespace topics (preview)
	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
	Create a data history connection for Azure Digital Twins
	Migrate an application to use passwordless connections with Azure Event Hubs
az eventhubs eventhub authorization-rule	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
	Export IoT data to Event Hubs
	Integrate Azure Digital Twins with Azure Time Series Insights
	Create and delete routes and endpoints by using the Azure CLI
	Extend Azure IoT Central with custom analytics using Azure Databricks
az eventhubs eventhub authorization-rule keys	Get an Event Hubs connection string

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
	Export IoT data to Event Hubs
	Integrate Azure Digital Twins with Azure Time Series Insights
	Create and delete routes and endpoints by using the Azure CLI
	Extend Azure IoT Central with custom analytics using Azure Databricks
<code>az eventhubs georecovery-alias authorization-rule keys</code>	Get an Event Hubs connection string
<code>az eventhubs namespace</code>	Quickstart: Send events to and receive events from Azure Event Hubs using .NET
	Send events to or receive events from event hubs by using Python
	Use Java to send events to or receive events from Azure Event Hubs
	Send events to or receive events from event hubs by using JavaScript
	Quickstart: Subscribe to Azure Kubernetes Service (AKS) events with Azure Event Grid
	Quickstart: Route custom events to Azure Event Hubs with Azure CLI and Event Grid
	Grant managed identity the access to Event Grid destination
	Quickstart: Create an event hub using Azure CLI
	Publish and deliver events using namespace topics (preview)
	Tutorial: Create a function in Java with an Event Hub trigger and an Azure Cosmos DB output binding
<code>az eventhubs namespace application-group</code>	Govern resources for client applications with application groups

Reference subgroup	Azure CLI article showing reference use
az eventhubs namespace application-group policy	Govern resources for client applications with application groups
az eventhubs namespace authorization-rule keys	Get an Event Hubs connection string
	Use Spring Kafka with Azure Event Hubs for Kafka API

az extension

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az extension	Install Azure CLI on Windows
	Use and manage extensions with the Azure CLI
	Azure CLI configuration
	Overview: Azure CLI terminology and support levels
	Available Azure CLI extensions
	Azure CLI interactive mode
	How to use the Azure CLI alias extension
	Migrate to workspace-based Application Insights resources
	Create a storage account
	Quickstart: Deploy your first application to Azure Spring Apps

az feature

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az feature	Use a public standard load balancer in Azure Kubernetes Service (AKS)
	Configure Azure CNI Overlay networking in Azure Kubernetes Service (AKS)

Reference subgroup	Azure CLI article showing reference use
	Use the Azure portal to enable end-to-end encryption using encryption at host
	Hibernating virtual machines
	Azure VM Image Builder overview
	Configure an AKS cluster
	Use Microsoft Entra pod-managed identities in Azure Kubernetes Service (Preview)
	Secure traffic between pods using network policies in Azure Kubernetes Service (AKS)
	Auto-upgrade node OS images
	Private Application Gateway deployment (preview)
az feature registration	Tutorial: Create a payment HSM
	Quickstart: Create an Azure Managed CCF resource using the Azure SDK for JavaScript and TypeScript
	Register the Azure Payment HSM resource providers and resource provider features
	Quickstart: Create an Azure Payment HSM with the Azure CLI
	Tutorial: Create a payment HSM with host and management port in different virtual networks using ARM template
	Quickstart: Create an Azure Managed CCF resource using the Azure SDK for Python
	Quickstart: Create an Azure payment HSM using an ARM template
	Quickstart: Create an Azure Managed CCF resource using the Azure SDK for Java
	Create a payment HSM with host and management port with IP addresses in different virtual networks using ARM template
	Quickstart: Create an Azure Managed CCF resource using the Azure SDK for Go

az feedback

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Install Azure CLI on Windows
	Install Azure CLI on macOS
	Get started with Azure CLI

az find

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Get started with Azure CLI
	Manage an App Service Environment
	Tutorial: Order Azure Data Box
	Manage your Azure AI Search service with the Azure CLI

az fleet

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az fleet	Quickstart: Create an Azure Kubernetes Fleet Manager resource and join member clusters
	Set up multi-cluster layer 4 load balancing across Azure Kubernetes Fleet Manager member clusters (preview)
az fleet member	Quickstart: Create an Azure Kubernetes Fleet Manager resource and join member clusters
	Orchestrate updates across multiple clusters by using Azure Kubernetes Fleet Manager
	Frequently Asked Questions - Azure Kubernetes Fleet Manager
az fleet updaterule	Orchestrate updates across multiple clusters by using Azure Kubernetes Fleet Manager

Reference subgroup	Azure CLI article showing reference use
az fleet updatestrategy	Orchestrate updates across multiple clusters by using Azure Kubernetes Fleet Manager

az functionapp

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az functionapp	App settings reference for Azure Functions
	Manage your function app
	Azure Functions Premium plan
	Quickstart: Create a Python function in Azure from the command line
	IP addresses in Azure Functions
	Quickstart: Create a C# function in Azure from the command line
	Tutorial: Use identity-based connections instead of secrets with triggers and bindings
	Create your first containerized functions on Azure Container Apps
	Azure Container Apps hosting of Azure Functions
	Use system-assigned managed identities to access Azure Cosmos DB data
az functionapp config	Guide for running C# Azure Functions in an isolated worker process
	Azure Functions Node.js developer guide
	Migrate apps from Azure Functions version 3.x to version 4.x
	How to target Azure Functions runtime versions
	Migrate apps from Azure Functions version 1.x to version 4.x
	Update language stack versions in Azure Functions

Reference subgroup	Azure CLI article showing reference use
az functionapp config appsettings	Manage your function app How to configure monitoring for Azure Functions
	 How to target Azure Functions runtime versions
	 Update language stack versions in Azure Functions
az functionapp config apptings	Guide for running C# Azure Functions in an isolated worker process
	 Manage your function app How to configure monitoring for Azure Functions
	 Azure Functions Node.js developer guide
	 Migrate apps from Azure Functions version 3.x to version 4.x
	 How to target Azure Functions runtime versions
	 Create your first durable function in Python
	 Azure Functions Java developer guide
	 How to disable functions in Azure Functions
	 Migrate apps from Azure Functions version 1.x to version 4.x
az functionapp config container	Working with containers and Azure Functions
	 Azure Container Apps hosting of Azure Functions
	 Deploy a machine learning model to Azure Functions with Azure Cache for Redis
az functionapp connection	Quickstart: Create a service connection in Azure Functions with the Azure CLI
az functionapp connection create	Tutorial: Python function with Azure Blob Storage as input
	 Tutorial: Python function with Azure Queue Storage as trigger
	 Tutorial: Python function with Azure Table Storage as output
	 Quickstart: Create a service connection in Azure Functions with the Azure CLI
az functionapp cors	Manage your function app

Reference subgroup	Azure CLI article showing reference use
az functionapp deployment	Manage your function app
az functionapp deployment container	Working with containers and Azure Functions
az functionapp deployment github-actions	Continuous delivery by using GitHub Actions
az functionapp deployment slot	Migrate apps from Azure Functions version 3.x to version 4.x Azure Functions deployment slots
	Migrate apps from Azure Functions version 1.x to version 4.x Automate tasks with Azure CLI
az functionapp deployment source	Zip deployment for Azure Functions Disable basic authentication in App Service deployments
	Tutorial Step 2: Automate resizing uploaded images using Event Grid
	Tutorial: Build out an end-to-end solution
	Tutorial: Deploy to Azure Functions using Jenkins
	Create a function app in Azure that is deployed from GitHub Automate tasks with Azure CLI
az functionapp function	Create your first containerized functions on Azure Container Apps Create your first containerized Azure Functions
	Tutorial: Build out an end-to-end solution
	Ingest IoT Hub telemetry into Azure Digital Twins Set up twin-to-twin event handling
az functionapp identity	Tutorial: Build out an end-to-end solution Ingest IoT Hub telemetry into Azure Digital Twins
	Integrate Azure Digital Twins with Azure Time Series Insights Set up twin-to-twin event handling

Reference subgroup	Azure CLI article showing reference use
az functionapp keys	Manually run a non HTTP-triggered function
az functionapp log deployment	Disable basic authentication in App Service deployments
az functionapp plan	Manage your function app
	Azure Functions Premium plan
	Create your first containerized Azure Functions
	Create a Function App in an App Service plan
	Create a function app in a Premium plan - Azure CLI

az grafana

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az grafana	Configure SMTP settings
	Set up Azure Managed Grafana authentication and permissions
	Set up private access (preview)
	Setup of Network Observability for Azure Kubernetes Service (AKS) - Azure managed Prometheus and Grafana
	Create and manage Grafana API keys in Azure Managed Grafana (Deprecated)
	Quickstart: Create an Azure Managed Grafana instance using the Azure CLI
	How to use service accounts in Azure Managed Grafana
	Use deterministic outbound IPs
	Enable zone redundancy in Azure Managed Grafana
az grafana api-key	Create and manage Grafana API keys in Azure Managed Grafana (Deprecated)
az grafana dashboard	Create a dashboard in Azure Managed Grafana
	Create and manage Grafana API keys in Azure Managed Grafana (Deprecated)

Reference subgroup	Azure CLI article showing reference use
az grafana data-source	How to manage data sources in Azure Managed Grafana Use deterministic outbound IPs
	 Configure an Azure Data Explorer datasource
az grafana service-account	How to use service accounts in Azure Managed Grafana
az grafana service-account token	How to use service accounts in Azure Managed Grafana

az graph

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az graph	Transfer an Azure subscription to a different Microsoft Entra directory Advanced Resource Graph query samples
	 Starter Resource Graph query samples
	 Get resource configuration changes
	 Azure Hybrid Benefit for Azure Stack HCI
	 Explore your Azure resources with Resource Graph
	 Working with large Azure resource data sets
	 Quickstart: Run your first Resource Graph query using Azure CLI
	 Upgrade public IP addresses attached to VM from Basic to Standard
	 Azure Arc extension management on Azure Stack HCI
az graph shared-query	 Quickstart: Create a Resource Graph shared query using Azure CLI

az group

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az group	Tips for using the Azure CLI successfully Learn to use Bash with the Azure CLI How to use variables in Azure CLI commands How to manage Azure resource groups with the Azure CLI Prepare your environment for the Azure CLI Overview and Prerequisites Create a resource using a service principal Delete Azure resources at scale using a script Cleanup & troubleshoot service principals Cleanup
az group deployment	Quickstart: Configure Microsoft Dev Box by using an ARM template

az hdinsight

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az hdinsight	Manage logs for an HDInsight cluster Manage Azure HDInsight clusters using Azure CLI Azure HDInsight double encryption for data at rest Troubleshoot a slow or failing job on a HDInsight cluster Tutorial: Create an Apache Kafka REST proxy enabled cluster in HDInsight using Azure CLI Create a cluster with Data Lake Storage Gen2 using Azure CLI Delete an HDInsight cluster using your browser, PowerShell, or the Azure CLI Quickstart: Create Apache Spark cluster in Azure HDInsight using Azure CLI Azure HDInsight: Azure CLI samples

Reference subgroup	Azure CLI article showing reference use
	Create HDInsight clusters using the Azure CLI
az hdinsight application	Azure HDInsight: Azure CLI samples
	Install custom Apache Hadoop applications on Azure HDInsight
az hdinsight azure-monitor	Use Azure Monitor logs to monitor HDInsight clusters
az hdinsight monitor	Use Azure Monitor logs to monitor HDInsight clusters
az hdinsight script-action	Customize Azure HDInsight clusters by using script actions
	Azure HDInsight: Azure CLI samples

az healthcareapis

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az healthcareapis acr	Converting your data to FHIR for Azure API for FHIR
az healthcareapis service	Configure customer-managed keys at rest
	Quickstart: Use an ARM template to deploy Azure API for FHIR

az hpc-cache

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az hpc-cache	Integrate Azure HPC Cache with Azure Kubernetes Service (AKS)
	Create an Azure HPC Cache
	Manage your cache
az hpc-cache blob-storage-target	Integrate Azure HPC Cache with Azure Kubernetes Service (AKS)

Reference subgroup	Azure CLI article showing reference use
	Add storage targets
	Edit storage targets
az hpc-cache nfs-storage-target	Add storage targets
	Edit storage targets
az hpc-cache skus	Create an Azure HPC Cache
az hpc-cache storage-target	Add storage targets
	View and manage storage targets
az hpc-cache usage-model	Add storage targets
	Edit storage targets

az identity

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az identity	Use Key Vault references as app settings in Azure App Service and Azure Functions
	How to use managed identities for App Service and Azure Functions
	Manage user-assigned managed identities
	Transfer an Azure subscription to a different Microsoft Entra directory
	Use a managed identity in Azure Kubernetes Service (AKS)
	Configure a custom container for Azure App Service
	Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)
	Migrate custom software to Azure App Service using a custom container
	Provision and publish a bot
	Use kubenet networking with your own IP address ranges in Azure Kubernetes Service (AKS)
az identity federated-	Connect your Azure identity provider to the Azure Key Vault Secrets

Reference subgroup	Azure CLI article showing reference use
credential	Store CSI Driver in Azure Kubernetes Service (AKS) Tutorial: Use a workload identity with an application on Azure Kubernetes Service (AKS)
	Deploy and configure workload identity on an Azure Kubernetes Service (AKS) cluster
	Tutorial: Deploy applications using GitOps with Flux v2
	Configure a user-assigned managed identity to trust an external identity provider
	Migrate from pod managed-identity to workload identity
	Quickstart: Deploy Application Gateway for Containers ALB Controller
	Azure App Configuration Kubernetes Provider reference
	Deploy an AI model on Azure Kubernetes Service (AKS) with the AI toolchain operator (preview)
	Integrate KEDA with your Azure Kubernetes Service cluster

az image

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az image	Azure Virtual Machine Scale Set agents Create a legacy managed image of a generalized VM in Azure
	Create an Azure Image Builder Bicep or ARM template JSON template
	How to use Packer to create Linux virtual machine images in Azure
	Disable or remove the Linux Agent from VMs and images
	Use the Azure Compute Gallery to create a custom image pool
	Creating generalized images without a provisioning agent
az image builder	Troubleshoot Azure VM Image Builder How to use Azure Image Builder triggers to set up an automatic image build

Reference subgroup	Azure CLI article showing reference use
az image builder identity	Troubleshoot Azure VM Image Builder
az image builder trigger	How to use Azure Image Builder triggers to set up an automatic image build

az import-export

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az import-export	Tutorial: Transfer data to Azure Files with Azure Import/Export
	Tutorial: Import data to Blob Storage with Azure Import/Export service
	Tutorial: Export data from Azure Blob storage with Azure Import/Export
az import-export location	Tutorial: Transfer data to Azure Files with Azure Import/Export
	Tutorial: Import data to Blob Storage with Azure Import/Export service
	Tutorial: Export data from Azure Blob storage with Azure Import/Export

az init

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Azure CLI configuration
	Prepare your environment for the Azure CLI

az interactive

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Get started with Azure CLI
	Overview and Prerequisites
	Azure CLI interactive mode
	Install Azure CLI on Azure Stack Hub
	Manage and deploy resources to Azure Stack Hub with Azure CLI - Modular Data Center (MDC)
	Manage and deploy resources to Azure Stack Hub with Azure CLI

az internet-analyzer

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az internet-analyzer	Create an Internet Analyzer test using CLI (Preview)
az internet-analyzer preconfigured-endpoint	Create an Internet Analyzer test using CLI (Preview)
az internet-analyzer profile	Create an Internet Analyzer test using CLI (Preview)
az internet-analyzer test	Embed the Internet Analyzer client
	Create an Internet Analyzer test using CLI (Preview)

az iot

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az iot central app	Extend Azure IoT Central with custom analytics using Azure Databricks
	Manage IoT Central from Azure CLI or PowerShell
az iot central app identity	Export IoT data to Blob Storage
	Export IoT data to Event Hubs
	Export IoT data to Azure Data Explorer

Reference subgroup	Azure CLI article showing reference use Export IoT data to Service Bus
	Extend Azure IoT Central with custom analytics using Azure Databricks
	Manage IoT Central from Azure CLI or PowerShell
az iot central device	Device authentication concepts in IoT Central
	Device implementation and best practices for IoT central
az iot central device twin	Monitor device connectivity using Azure CLI
az iot central diagnostics	Monitor device connectivity using Azure CLI
az iot device	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Quickstart: Deploy an Azure IoT hub and a storage account using Bicep
	Tutorial: Send email notifications about Azure IoT Hub events using Event Grid and Logic Apps
	Quickstart: Deploy an Azure IoT hub and a storage account using an ARM template
	Get started with device management (Azure CLI)
	Tutorial: Automate Azure Device Provisioning Service with GitHub Actions
	Get started with device twins (Azure CLI)
	Schedule and broadcast jobs (Azure CLI)
az iot device c2d-message	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Tutorial - Use MQTT to develop an IoT device client without using a device SDK
az iot device registration	Tutorial: Automate Azure Device Provisioning Service with GitHub Actions
az iot dps	Tutorial: Set up your environment for the IoT Plug and Play quickstarts and tutorials
	Tutorial: Use custom allocation policies with Device Provisioning Service (DPS)

Reference subgroup	Azure CLI article showing reference use
	Automange devices in Azure Digital Twins using Device Provisioning Service (DPS)
	Quickstart: Use Terraform to create an Azure IoT Device Provisioning Service
	How to link and manage IoT hubs
	How to use allocation policies to provision devices across IoT hubs
	Quickstart: Set up the IoT Hub Device Provisioning Service with Azure CLI
	How to use symmetric keys over HTTPS without an SDK
	Quickstart: Set up the IoT Hub Device Provisioning Service (DPS) with an ARM template
	Quickstart: Set up the IoT Hub Device Provisioning Service (DPS) with Bicep
az iot dps certificate	How to use X.509 certificates over HTTPS without an SDK
az iot dps enrollment	Tutorial: Set up your environment for the IoT Plug and Play quickstarts and tutorials
	How to use X.509 certificates over HTTPS without an SDK
	How to use allocation policies to provision devices across IoT hubs
	How to use symmetric keys over HTTPS without an SDK
	Tutorial: Automate Azure Device Provisioning Service with GitHub Actions
az iot dps enrollment-group	Tutorial: Provision devices using symmetric key enrollment groups
	Symmetric key attestation
	Tutorial: Use custom allocation policies with Device Provisioning Service (DPS)
	How to use X.509 certificates over HTTPS without an SDK
	How to use allocation policies to provision devices across IoT hubs
	How to use symmetric keys over HTTPS without an SDK

Reference subgroup	Azure CLI article showing reference use
az iot dps linked-hub	Tutorial: Set up your environment for the IoT Plug and Play quickstarts and tutorials
	Tutorial: Use custom allocation policies with Device Provisioning Service (DPS)
	How to link and manage IoT hubs
	Quickstart: Set up the IoT Hub Device Provisioning Service with Azure CLI
	Tutorial: Automate Azure Device Provisioning Service with GitHub Actions
az iot du account	Create Device Update for IoT Hub resources
	Configure access control roles for Device Update resources
	Configure private endpoints for Device Update for IoT Hub accounts
az iot du account private-endpoint-connection	Configure private endpoints for Device Update for IoT Hub accounts
az iot du device	Find and fix devices missing from Device Update for IoT Hub using agent check
az iot du device deployment	Deploy an update by using Device Update for Azure IoT Hub
az iot du device group	Manage device groups in Device Update for IoT Hub
	Deploy an update by using Device Update for Azure IoT Hub
az iot du device health	Find and fix devices missing from Device Update for IoT Hub using agent check
az iot du device log	Remotely collect diagnostic logs from devices using Device Update for IoT Hub
az iot du instance	Create Device Update for IoT Hub resources
	Remotely collect diagnostic logs from devices using Device Update for IoT Hub
az iot du update	Import an update to Device Update for IoT Hub
az iot du update init	Prepare an update to import into Device Update for IoT Hub
	How to understand and use delta updates in Device Update for IoT Hub (Preview)
	Use the related files feature to reference multiple update files

Reference subgroup	Azure CLI article showing reference use
az iot edge	Tutorial: Develop IoT Edge modules using Visual Studio Code Install and run the Spatial Analysis container (preview) Use Visual Studio 2022 to develop and debug modules for Azure IoT Edge
	Connect Azure IoT Edge devices to create a hierarchy Quickstart: Deploy a connected registry to an IoT Edge device Debug Azure IoT Edge modules using Visual Studio Code
	Deploy a Spatial Analysis web application Tutorial: Develop IoT Edge modules with Linux containers using IoT Edge for Linux on Windows
	Deploy Azure IoT Edge modules with Azure CLI Tutorial: Deploy a connected registry to a nested IoT Edge hierarchy
az iot edge deployment	Monitor IoT Edge deployments Deploy and monitor IoT Edge modules at scale by using the Azure CLI
az iot edge devices	Tutorial: Create a hierarchy of IoT Edge devices Tutorial: Create a hierarchy of IoT Edge devices using IoT Edge for Linux on Windows
az iot hub	Overview: Azure CLI terminology and support levels Quickstart: Send telemetry from an IoT Plug and Play device to Azure IoT Hub Understand and invoke direct methods from IoT Hub
	Quickstart: Deploy your first IoT Edge module to a virtual Linux device Send cloud-to-device messages from an IoT hub
	Quickstart: Deploy your first IoT Edge module to a Windows device Install and run the Spatial Analysis container (preview)

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Quickstart: Connect an MXCHIP AZ3166 devkit to IoT Hub
	Tutorial - Use MQTT to develop an IoT device client without using a device SDK
az iot hub certificate root-authority	Migrate IoT Hub resources to a new TLS certificate root
az iot hub configuration	How to automatically migrate an IoT hub using the Azure CLI
	Automatic IoT device and module management using the Azure CLI
az iot hub connection-string	Quickstart: Send telemetry from an IoT Plug and Play device to Azure IoT Hub
	Tutorial: Send device data to Azure Storage using IoT Hub message routing
	Quickstart: Connect an MXCHIP AZ3166 devkit to IoT Hub
	Tutorial: Visualize real-time sensor data from your Azure IoT hub in a web application
	Tutorial: Build out an end-to-end solution
	Quickstart: Connect an ESPRESSIF ESP32-Azure IoT Kit to IoT Hub
	Tutorial: Configure your devices from a back-end service
	Tutorial: Set up your environment for the IoT Plug and Play quickstarts and tutorials
	Tutorial: Use custom allocation policies with Device Provisioning Service (DPS)
	How to link and manage IoT hubs
az iot hub consumer-group	Tutorial: Visualize real-time sensor data from your Azure IoT hub in a web application
az iot hub device-identity	Create and provision an IoT Edge device on Linux using symmetric keys
	Quickstart: Send telemetry from an IoT Plug and Play device to Azure IoT Hub

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Deploy your first IoT Edge module to a virtual Linux device
	Quickstart: Deploy your first IoT Edge module to a Windows device
	Tutorial: Send device data to Azure Storage using IoT Hub message routing
	Install and run the Spatial Analysis container (preview)
	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Tutorial - Use MQTT to develop an IoT device client without using a device SDK
	Tutorial: Build out an end-to-end solution
	Create and provision an IoT Edge for Linux on Windows device using symmetric keys
<code>az iot hub device-identity connection-string</code>	Create and provision an IoT Edge device on Linux using symmetric keys
	Quickstart: Send telemetry from an IoT Plug and Play device to Azure IoT Hub
	Quickstart: Deploy your first IoT Edge module to a virtual Linux device
	Quickstart: Deploy your first IoT Edge module to a Windows device
	Install and run the Spatial Analysis container (preview)
	Tutorial: Build out an end-to-end solution
	Use Visual Studio 2022 to develop and debug modules for Azure IoT Edge
	Create and provision an IoT Edge for Linux on Windows device using symmetric keys
	Tutorial: Use a simulated device to test connectivity with your IoT hub
	Tutorial: Configure your devices from a back-end service
<code>az iot hub devicestream</code>	IoT Hub Device Streams (preview)

Reference subgroup	Azure CLI article showing reference use
az iot hub device-twin	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Quickstart: Connect an MXCHIP AZ3166 devkit to IoT Hub
	Quickstart: Connect an ESPRESSIF ESP32-Azure IoT Kit to IoT Hub
	Tutorial: Use a simulated device to test connectivity with your IoT hub
	Tutorial: Use Azure IoT Hub message enrichments
	Manage device groups in Device Update for IoT Hub
	How to manage devices using device twin tags in Azure IoT Hub
	Quickstart: Connect an STMicroelectronics B-L475E-IOT01A Discovery kit to IoT Hub
	Quickstart: Connect a Microchip ATSAME54-XPro Evaluation kit to IoT Hub
	Quickstart: Connect an STMicroelectronics B-U585I-IOT02A Discovery kit to IoT Hub
az iot hub identity	Configure IoT Hub file uploads using Azure CLI
az iot hub job	Overview: Azure CLI terminology and support levels
	Schedule and broadcast jobs (Azure CLI)
az iot hub message-endpoint	Create and delete routes and endpoints by using the Azure CLI
az iot hub message-endpoint create	Create and delete routes and endpoints by using the Azure CLI
az iot hub message-enrichment	Tutorial: Use Azure IoT Hub message enrichments
az iot hub message-route	Create and delete routes and endpoints by using the Azure CLI
az iot hub message-route fallback	Create and delete routes and endpoints by using the Azure CLI
az iot hub module-identity	Deploy Azure IoT Edge modules with Azure CLI
	Get started with IoT Hub module identities and module twins using Azure CLI
az iot hub module-twin	Troubleshoot your IoT Edge device

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Create a hierarchy of IoT Edge devices
	Manage device groups in Device Update for IoT Hub
	Tutorial: Create a hierarchy of IoT Edge devices using IoT Edge for Linux on Windows
	Get started with IoT Hub module identities and module twins using Azure CLI
az iot hub policy	How to link and manage IoT hubs
	Use Apache Kafka on HDInsight with Azure IoT Hub
az iot hub route	Tutorial: Send device data to Azure Storage using IoT Hub message routing
	Tutorial: Use Azure IoT Hub message enrichments
az iot hub routing-endpoint	Tutorial: Send device data to Azure Storage using IoT Hub message routing
	Tutorial: Use Azure IoT Hub message enrichments
az iot hub state	How to automatically migrate an IoT hub using the Azure CLI
az iot ops	Quickstart: Deploy Azure IoT Operations to an Arc-enabled Kubernetes cluster
	Prepare your Azure Arc-enabled Kubernetes cluster
	Deploy Azure IoT Operations extensions to a Kubernetes cluster
	Manage secrets for your Azure IoT Operations deployment
az iot ops asset	Manage asset configurations remotely
az iot ops asset data-point	Manage asset configurations remotely
az iot ops asset endpoint	Manage asset configurations remotely
az iot ops asset event	Manage asset configurations remotely
az iot ops mq	Configure Azure IoT MQ authentication

az k8s-configuration

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az k8s-configuration	Tutorial: Deploy applications using GitOps with Flux v2 Application deployments with GitOps (Flux v2) for AKS and Azure Arc-enabled Kubernetes
	Tutorial: Implement CI/CD with GitOps using Azure Arc-enabled Kubernetes clusters
	Tutorial: Deploy configurations using GitOps on an Azure Arc-enabled Kubernetes cluster
	Troubleshoot extension issues for Azure Arc-enabled Kubernetes clusters
	Deploy Helm Charts using GitOps on an Azure Arc-enabled Kubernetes cluster
az k8s-configuration flux	Tutorial: Deploy applications using GitOps with Flux v2 Tutorial: Implement CI/CD with GitOps (Flux v2)
	GitOps (Flux v2) supported parameters
	Troubleshoot extension issues for Azure Arc-enabled Kubernetes clusters
az k8s-configuration flux kustomization	GitOps (Flux v2) supported parameters

az k8s-extension

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az k8s-extension	Enable monitoring for Kubernetes clusters Understand Azure Policy for Kubernetes clusters
	Configure Microsoft Defender for Containers components
	Tutorial: Deploy applications using GitOps with Flux v2
	Disable monitoring of your Kubernetes cluster

Reference subgroup	Azure CLI article showing reference use
	Application deployments with GitOps (Flux v2) for AKS and Azure Arc-enabled Kubernetes
	Deploy Azure Machine Learning extension on AKS or Arc Kubernetes cluster
	Manage Azure Kubernetes Service backups using Azure Backup
	Reference for configuring Kubernetes cluster for Azure Machine Learning
	Configure data collection in Container insights using data collection rule
az k8s-extension extension-types	Deploy and manage applications from Azure Marketplace on Azure Arc-enabled Kubernetes clusters

az keyvault

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az keyvault	Populate variables for use in scripts
	Assign a Key Vault access policy (legacy)
	Use the Azure Key Vault provider for Secrets Store CSI Driver in an Azure Kubernetes Service (AKS) cluster
	Azure Key Vault configuration provider in ASP.NET Core
	Transfer an Azure subscription to a different Microsoft Entra directory
	Use Azure Key Vault secrets in Azure Pipelines
	Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)
	Quickstart: Azure Key Vault secret client library for .NET
	Quickstart: Azure Key Vault secret client library for Python
	Tutorial: Use a managed identity to connect Key Vault to an Azure web app in .NET
az keyvault backup	Full backup and restore and selective key restore

Reference subgroup	Azure CLI article showing reference use
	Managed HSM disaster recovery
az keyvault certificate	Azure Key Vault recovery management with soft delete and purge protection
	Tutorial: Import a certificate in Azure Key Vault
	Azure Key Vault backup and restore
	Export certificates from Azure Key Vault
	Renew your Azure Key Vault certificates
	Set up a custom domain name and SSL certificate with the application routing add-on
	Tutorial: Use TLS/SSL certificates to secure a web server
	Tutorial - How to use cloud-init to customize a Linux virtual machine in Azure on first boot
	Set up Secrets Store CSI Driver to enable NGINX Ingress Controller with TLS
	Manage Key Vault using the Azure CLI
az keyvault key	Azure Key Vault recovery management with soft delete and purge protection
	Configure cryptographic key auto-rotation in Azure Key Vault
	Azure Key Vault backup and restore
	Configure customer-managed keys in the same tenant for an existing storage account
	Quickstart: Create an Azure key vault and key using Terraform
	Create and configure a key vault for Azure Disk Encryption on a Windows VM
	Manage Key Vault using the Azure CLI
	Azure OpenAI Service encryption of data at rest
	Add Key Management Service etcd encryption to an Azure Kubernetes Service cluster
	Configure customer-managed keys for your Azure Cosmos DB account with Azure Key Vault

Reference subgroup	Azure CLI article showing reference use
az keyvault key rotation-policy	Configure cryptographic key auto-rotation in Azure Key Vault
	Configure key auto-rotation in Azure Managed HSM
az keyvault network-rule	Troubleshooting online endpoints deployment and scoring
	Configure Azure Key Vault networking settings
	Secure your managed online endpoints with network isolation
az keyvault private-endpoint-connection	Integrate Key Vault with Azure Private Link
	Integrate Managed HSM with Azure Private Link
az keyvault region	Enable multi-region replication on Azure Managed HSM
az keyvault restore	Full backup and restore and selective key restore
	Managed HSM disaster recovery
az keyvault role assignment	Use the Azure CLI to enable end-to-end encryption using encryption at host
	Configure encryption with customer-managed keys stored in Azure Key Vault Managed HSM
	Managed HSM role management
	Secure access to your managed HSMs
	Use the Azure CLI to enable server-side encryption with customer-managed keys for managed disks
	Azure Managed HSM TLS Offload Library
	Quickstart: Deploy confidential VM with ARM template
	Configure customer-managed keys for your Azure Cosmos DB account with Azure Managed HSM Key Vault
	Configure HSM customer-managed keys for Azure managed disks
	Integrate Azure Managed HSM with Azure Policy
az keyvault role definition	Managed HSM role management
	Azure Managed HSM TLS Offload Library
az keyvault secret	Output formats for Azure CLI commands

Reference subgroup	Azure CLI article showing reference use
	Get an existing service principal
	Use an Azure service principal with certificate-based authentication
	Populate variables for use in scripts
	Use the Azure Key Vault provider for Secrets Store CSI Driver in an Azure Kubernetes Service (AKS) cluster
	Azure Key Vault configuration provider in ASP.NET Core
	Use Azure Key Vault secrets in Azure Pipelines
	Store Terraform state in Azure Storage
	Quickstart: Azure Key Vault secret client library for Python
	Azure Key Vault recovery management with soft delete and purge protection
az keyvault security-domain	Quickstart: Provision and activate a Managed HSM using Azure CLI
	Managed HSM disaster recovery
az keyvault storage	Manage storage account keys with Key Vault and the Azure CLI (legacy)
az keyvault storage sas-definition	Manage storage account keys with Key Vault and the Azure CLI (legacy)

az kusto

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az kusto cluster	Create an Azure Data Explorer cluster and database
	Create a data history connection for Azure Digital Twins
	Create business continuity and disaster recovery solutions with Azure Data Explorer
	Configure customer-managed keys
az kusto database	Create an Azure Data Explorer cluster and database

Reference subgroup	Azure CLI article showing reference use
	Create a data history connection for Azure Digital Twins
az kusto database-principal-assignment	Export IoT data to Azure Data Explorer

az lab

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az lab vm	Quickstart: Create a lab in Azure DevTest Labs using Terraform
	Create and manage virtual machines with DevTest Labs using the Azure CLI
	Azure CLI Samples for Azure DevTest Labs

az load

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az load	Load test a website by using a JMeter script in Azure Load Testing
	Tutorial: Run a load test to identify performance bottlenecks in a web app
	Use managed identities for Azure Load Testing
az load test	Load test a website by using a JMeter script in Azure Load Testing
	Tutorial: Run a load test to identify performance bottlenecks in a web app
	Manually configure your CI/CD workflow for running load tests
az load test-run	Load test a website by using a JMeter script in Azure Load Testing
	Manually configure your CI/CD workflow for running load tests
az load test-run metrics	Load test a website by using a JMeter script in Azure Load Testing

Reference subgroup	Azure CLI article showing reference use
Manually configure your CI/CD workflow for running load tests	

az lock

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az lock	Lock your resources to protect your infrastructure Manage Azure Resource Groups by using Azure CLI Apply an Azure Resource Manager lock to a storage account How to protect private DNS zones and records Use the in-place migration feature to migrate App Service Environment v1 and v2 to App Service Environment v3 How to protect DNS zones and records Automatic deletions from deployment history Manage Azure Cosmos DB for NoSQL resources using Azure CLI Manage Azure resources by using Azure CLI Protect Azure Cosmos DB resources with locks

az logic

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az logic integration-account	Create and manage integration accounts for B2B workflows in Azure Logic Apps with the Enterprise Integration Pack
az logic workflow	Quickstart: Create and deploy a Consumption logic app workflow in multi-tenant Azure Logic Apps with an ARM template
	Quickstart: Create and manage workflows with Azure CLI in Azure Logic Apps

Reference subgroup	Azure CLI article showing reference use
	Azure CLI script sample - create a logic app

az logicapp

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az logicapp	Create and deploy single-tenant based logic app workflows with Azure Arc-enabled Logic Apps (Preview)
az logicapp config appsettings	Edit host and app settings for Standard logic apps in single-tenant Azure Logic Apps
	Create cross-environment parameters for workflow inputs in Azure Logic Apps
	Create and deploy single-tenant based logic app workflows with Azure Arc-enabled Logic Apps (Preview)
az logicapp config apptings	Edit host and app settings for Standard logic apps in single-tenant Azure Logic Apps
	Create cross-environment parameters for workflow inputs in Azure Logic Apps
	Create and deploy single-tenant based logic app workflows with Azure Arc-enabled Logic Apps (Preview)
az logicapp deployment source	Set up DevOps deployment for Standard logic app workflows in single-tenant Azure Logic Apps
	Create and deploy single-tenant based logic app workflows with Azure Arc-enabled Logic Apps (Preview)

az login

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	How to manage Azure subscriptions with the Azure CLI
	Get started with Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Create an Azure service principal with Azure CLI
	Sign in interactively with Azure CLI
	Sign in with a service principal using Azure CLI
	Sign in with a managed identity using Azure CLI
	How to use variables in Azure CLI commands
	Use an Azure service principal with certificate-based authentication
	Use an Azure service principal with password-based authentication
	Prepare your environment for the Azure CLI

az logout

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	How to manage Azure subscriptions with the Azure CLI
	Deploy to App Service using GitHub Actions
	Continuous deployment with custom containers in Azure App Service
	Use GitHub Actions workflow to deploy your static website in Azure Storage
	Quickstart: Create a policy assignment to identify non-compliant resources using Azure CLI
	Tutorial: Create a highly available multi-region app in Azure App Service
	Tutorial: Create a secure n-tier app in Azure App Service
	Quickstart: Create a policy assignment to identify non-compliant resources by using a Bicep file
	Use GitHub Actions to connect to Azure SQL Database
	Deploy a custom container to App Service using GitHub Actions

az maintenance

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az maintenance applyupdate	Control updates with Maintenance Configurations and the Azure CLI
az maintenance assignment	How to programmatically manage updates for Azure VMs Tutorial: Schedule updates on Dynamic scopes
	How to programmatically manage updates for Azure Arc-enabled servers
	Control updates with Maintenance Configurations and the Azure CLI
	Maintenance control for OS image upgrades on Azure Virtual Machine Scale Sets using Azure CLI
az maintenance configuration	How to programmatically manage updates for Azure VMs How to programmatically manage updates for Azure Arc-enabled servers
	Control updates with Maintenance Configurations and the Azure CLI
	Maintenance control for OS image upgrades on Azure Virtual Machine Scale Sets using Azure CLI
az maintenance public-configuration	Configure maintenance window in Azure SQL Database Configure maintenance window in Azure SQL Managed Instance
az maintenance update	Control updates with Maintenance Configurations and the Azure CLI

az managedapp

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az managedapp	Quickstart: Deploy a service catalog managed application
	Tutorial: Create managed application with custom actions and

Reference subgroup	Azure CLI article showing reference use
	resources
	Work with resources in the managed resource group for Azure managed application
	Quickstart: Use Bicep to deploy an Azure Managed Application definition
az managedapp definition	Quickstart: Create and publish an Azure Managed Application definition
	Quickstart: Deploy a service catalog managed application
	Quickstart: Use Bicep to create and publish an Azure Managed Application definition
	Tutorial: Create managed application with custom actions and resources
	Quickstart: Use Bicep to deploy an Azure Managed Application definition

az managed-cassandra

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az managed-cassandra cluster	Quickstart: Create an Azure Managed Instance for Apache Cassandra cluster from the Azure portal
	How to enable LDAP authentication in Azure Managed Instance for Apache Cassandra
	Quickstart: Create an Azure Managed Instance for Apache Cassandra cluster using Azure CLI
	Quickstart: Create a multi-region cluster with Azure Managed Instance for Apache Cassandra
	DBA commands for Azure Managed Instance for Apache Cassandra
	Manage Azure Managed Instance for Apache Cassandra resources using Azure CLI
	Customer-managed keys - overview

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Configure a hybrid cluster with Azure Managed Instance for Apache Cassandra
	Use VPN with Azure Managed Instance for Apache Cassandra
az managed-cassandra datacenter	How to enable LDAP authentication in Azure Managed Instance for Apache Cassandra
	Quickstart: Create an Azure Managed Instance for Apache Cassandra cluster using Azure CLI
	Quickstart: Create a multi-region cluster with Azure Managed Instance for Apache Cassandra
	Manage Azure Managed Instance for Apache Cassandra resources using Azure CLI
	Customer-managed keys - overview
	Quickstart: Configure a hybrid cluster with Azure Managed Instance for Apache Cassandra
	Materialized views in Azure Managed Instance for Apache Cassandra

az managedservices

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az managedservices assignment	Onboard a customer to Azure Lighthouse
	Remove access to a delegation
az managedservices definition	Onboard a customer to Azure Lighthouse

az managementpartner

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az managementpartner	Link a partner ID to your account that's used to manage customers

Reference subgroup	Azure CLI article showing reference use
	Link a partner ID to your Power Platform and Dynamics Customer Insights accounts

az maps

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az maps account keys	Create your Azure Maps account using an ARM template

az mariadb

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az mariadb server	Configure SSL connectivity in your application to securely connect to Azure Database for MariaDB
	Quickstart: Create an Azure Database for MariaDB server by using the Azure CLI
	Tutorial: Design an Azure Database for MariaDB using Azure CLI
	How to back up and restore a server in Azure Database for MariaDB using the Azure CLI
	Create and manage Private Link for Azure Database for MariaDB using CLI
	Create and manage Azure Database for MariaDB VNet service endpoints using Azure CLI
	List and update configurations of an Azure Database for MariaDB server using Azure CLI
	Create and manage Azure Database for MariaDB firewall rules by using the Azure CLI
	How to create and manage read replicas in Azure Database for MariaDB using the Azure CLI and REST API
	Monitor and scale an Azure Database for MariaDB server using Azure CLI

Reference subgroup	Azure CLI article showing reference use
az mariadb server configuration	Configure server parameters in Azure Database for MariaDB using the Azure CLI
	List and update configurations of an Azure Database for MariaDB server using Azure CLI
	Enable and download server slow query logs of an Azure Database for MariaDB server using Azure CLI
	Configure and access Azure Database for MariaDB slow query logs by using Azure CLI
	Configure and access Azure Database for MariaDB audit logs in the Azure CLI
az mariadb server firewall-rule	Quickstart: Create an Azure Database for MariaDB server by using the Azure CLI
	Tutorial: Design an Azure Database for MariaDB using Azure CLI
	Create and manage Azure Database for MariaDB firewall rules by using the Azure CLI
	Create a MariaDB server and configure a firewall rule using the Azure CLI
az mariadb server replica	How to create and manage read replicas in Azure Database for MariaDB using the Azure CLI and REST API
az mariadb server vnet-rule	Create and manage Azure Database for MariaDB VNet service endpoints using Azure CLI
	Create a MariaDB server and configure a vNet rule using the Azure CLI
az mariadb server-logs	Enable and download server slow query logs of an Azure Database for MariaDB server using Azure CLI
	Configure and access Azure Database for MariaDB slow query logs by using Azure CLI

az ml

[] Expand table

Reference subgroup	Azure CLI article showing reference use
az ml batch-endpoint	Create jobs and input data for batch endpoints Authorization on batch endpoints
	Using low priority VMs in batch deployments Image processing with batch model deployments
	Deploy language models in batch endpoints
az ml component	Create and run machine learning pipelines using components with the Azure Machine Learning CLI Share models, components, and environments across workspaces with registries
az ml compute	Troubleshooting online endpoints deployment and scoring Create an Azure Machine Learning compute instance
	Create an Azure Machine Learning compute cluster Train models with Azure Machine Learning CLI, SDK, and REST API
	Set up authentication between Azure Machine Learning and other services Set up AutoML to train a time-series forecasting model with SDK and CLI
	Create and run machine learning pipelines using components with the Azure Machine Learning CLI Secure an Azure Machine Learning training environment with virtual networks
	How Azure Machine Learning works: resources and assets Tutorial: Train an object detection model with AutoML and Python
az ml connection	Set up authentication between Azure Machine Learning and other services Create connections (preview)
az ml data	Create and manage data assets Working with tables in Azure Machine Learning
	Tutorial: Train an object detection model with AutoML and Python Create jobs and input data for batch endpoints

Reference subgroup	Azure CLI article showing reference use
	CLI (v2) MLtable YAML schema
	Import data assets (preview)
	Prepare data for computer vision tasks with automated machine learning
	Share data across workspaces with registries (preview)
	Manage imported data assets (preview)
az ml datastore	Create datastores
	How Azure Machine Learning works: resources and assets
	Create jobs and input data for batch endpoints
	Install & use the CLI (v1)
az ml environment	Deploy and score a machine learning model by using an online endpoint
	Manage Azure Machine Learning environments with the CLI & SDK (v2)
	Set up authentication between Azure Machine Learning and other services
	Share models, components, and environments across workspaces with registries
	Create & use software environments in Azure Machine Learning with CLI v1
	Install & use the CLI (v1)
	Customize environment for runtime
	Perform safe rollout of new deployments for real-time inference
az ml feature-set	Tutorial 1: Develop and register a feature set with managed feature store
	Feature set materialization concepts
az ml job	Create and manage data assets
	Access data in a job
	Work with models in Azure Machine Learning
	Set up AutoML training for tabular data with the Azure Machine Learning CLI and Python SDK
	Working with tables in Azure Machine Learning

Reference subgroup	Azure CLI article showing reference use
	Train models with Azure Machine Learning CLI, SDK, and REST API
	Git integration for Azure Machine Learning
	Set up AutoML to train a time-series forecasting model with SDK and CLI
	Manage inputs and outputs of component and pipeline
	Track ML experiments and models with MLflow
az ml model	Deploy and score a machine learning model by using an online endpoint
	Troubleshooting online endpoints deployment and scoring
	Work with models in Azure Machine Learning
	Tutorial: Train an object detection model with AutoML and Python
	Deploy MLflow models to online endpoints
	Share models, components, and environments across workspaces with registries
	Deploy machine learning models to Azure
	Install & use the CLI (v1)
	Update a deployed web service (v1)
	Set up AutoML to train computer vision models
az ml online-deployment	Deploy and score a machine learning model by using an online endpoint
	Troubleshooting online endpoints deployment and scoring
	Deploy models from HuggingFace hub to Azure Machine Learning online endpoints for real-time inference
	Authenticate clients for online endpoints
	Tutorial: Train an object detection model with AutoML and Python
	Deploy MLflow models to online endpoints
	Share models, components, and environments across workspaces with registries
	Set up AutoML to train computer vision models
	Secure your managed online endpoints with network isolation

Reference subgroup	Azure CLI article showing reference use
	Collect production data from models deployed for real-time inferencing (preview)
az ml online-endpoint	<p>Deploy and score a machine learning model by using an online endpoint</p> <p>Troubleshooting online endpoints deployment and scoring</p>
	Deploy models from HuggingFace hub to Azure Machine Learning online endpoints for real-time inference
	Authenticate clients for online endpoints
	Tutorial: Train an object detection model with AutoML and Python
	Deploy MLflow models to online endpoints
	Share models, components, and environments across workspaces with registries
	Set up AutoML to train computer vision models
	Secure your managed online endpoints with network isolation
	Use a custom container to deploy a model to an online endpoint
az ml registry	<p>Manage Azure Machine Learning registries</p> <p>Network isolation with Azure Machine Learning registries</p>
az ml schedule	<p>Monitor performance of models deployed to production</p> <p>Schedule data import jobs (preview)</p>

az mobile-network

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az mobile-network	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network attached-data-network	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network data-network	Quickstart: Deploy a private mobile network and site - Azure CLI

Reference subgroup	Azure CLI article showing reference use
az mobile-network pccp	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network pcdp	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network service	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network sim	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network sim group	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network sim policy	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network site	Quickstart: Deploy a private mobile network and site - Azure CLI
az mobile-network slice	Quickstart: Deploy a private mobile network and site - Azure CLI

az monitor

[\[\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az monitor account	Manage an Azure Monitor workspace
az monitor action-group	Switch to Azure Monitor based alerts for Azure Backup
	Quickstart: Create a budget with Bicep
	Create metric alert in Azure CLI
az monitor activity-log	Set up staging environments in Azure App Service
	View activity logs for Azure RBAC changes
	Troubleshoot Azure Database for PostgreSQL - Flexible Server CLI errors
	Troubleshoot Azure Database for MySQL - Flexible Server CLI errors

Reference subgroup	Azure CLI article showing reference use
az monitor activity-log alert	Create a new alert rule using the CLI, PowerShell, or an ARM template
	Quickstart: Create activity log alerts on service notifications using a Bicep file
	Quickstart: Create activity log alerts on service notifications using an ARM template
	Quickstart: Create Azure Advisor alerts on new recommendations using an ARM template
az monitor alert-processing-rule	Alert processing rules
	Switch to Azure Monitor based alerts for Azure Backup
az monitor app-insights component	Migrate to workspace-based Application Insights resources
	Workspace-based Application Insights resources
	Configure a Java app for Azure App Service
	Manage Azure Machine Learning workspaces using Azure CLI extension v1
	Manage Azure Machine Learning workspaces using Azure CLI
	Create Express.js virtual machine using Azure CLI
	Quickstart: Monitor applications end-to-end
az monitor app-insights component ed-storage	Configure BYOS for Application Insights Profiler and Snapshot Debugger
az monitor autoscale	Use autoscale actions to send email and webhook alert notifications in Azure Monitor
	Tutorial: Automatically scale a Virtual Machine Scale Set with the Azure CLI
	Configure VMware Spring Cloud Gateway
	Set up autoscale for applications
az monitor autoscale profile	Autoscale with multiple profiles
az monitor autoscale rule	Tutorial: Automatically scale a Virtual Machine Scale Set with the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Autoscale with multiple profiles
	Configure VMware Spring Cloud Gateway
	Set up autoscale for applications
az monitor data-collection rule	Data collection rules in Azure Monitor
	Create and edit data collection rules (DCRs) in Azure Monitor
	Azure Monitor managed service for Prometheus remote write
	Microsoft Entra authorization proxy
	Prometheus Remote-Write to Azure Monitor Workspace
	Monitor Nexus Kubernetes cluster
	Monitoring virtual machines (for virtualized network function)
az monitor data-collection rule association	Data collection rules in Azure Monitor
	Monitoring virtual machines (for virtualized network function)
az monitor diagnostic-settings	Azure Spring Apps FAQ
	Monitor Azure Firewall logs (legacy) and metrics
	Enable Key Vault logging
	Create diagnostic settings in Azure Monitor
	Monitor Azure Cosmos DB data by using diagnostic settings in Azure
	Monitoring load balancer
	Configure diagnostic log delivery
	Monitoring-áApp-áConfiguration
	Monitoring Public IP addresses
	Resource logging for a network security group
az monitor log-analytics	Monitor logs in Azure Container Apps with Log Analytics
	Tutorial: Deploy an event-driven job with Azure Container Apps
	Create a job with Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Deploy a Dapr application to Azure Container Apps using the Azure CLI
	Tutorial: Deploy a background processing application with Azure Container Apps
	Tutorial: Deploy a Dapr application to Azure Container Apps with an Azure Resource Manager or Bicep template
	Quickstart: Deploy an existing container image with the command line
az monitor log-analytics cluster	Create and manage a dedicated cluster in Azure Monitor Logs
	Azure Monitor customer-managed key
	Bring your own Machine Learning (ML) into Microsoft Sentinel
az monitor log-analytics workspace	Create a Log Analytics workspace
	Create and manage a dedicated cluster in Azure Monitor Logs
	Monitoring load balancer
	Delete and recover an Azure Log Analytics workspace
	Log storage and monitoring options in Azure Container Apps
	Monitoring Public IP addresses
	Tutorial: Troubleshoot an App Service app with Azure Monitor
	Connect Azure Arc-enabled servers to Microsoft Defender for Cloud
	Connect Azure Arc-enabled servers to Microsoft Sentinel
	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace data-export	Log Analytics workspace data export in Azure Monitor
	Bring your own Machine Learning (ML) into Microsoft Sentinel
	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace linked-service	Create and manage a dedicated cluster in Azure Monitor Logs
	Managing Azure Monitor Logs in Azure CLI

Reference subgroup	Azure CLI article showing reference use
az monitor log-analytics workspace linked-storage	Azure Monitor customer-managed key
	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace pack	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace saved-search	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace table	Data retention and archive in Azure Monitor Logs
	Set a table's log data plan to Basic or Analytics
	Manage tables in a Log Analytics workspace
	Run search jobs in Azure Monitor
	Managing Azure Monitor Logs in Azure CLI
az monitor log-analytics workspace table restore	Restore logs in Azure Monitor
az monitor log-analytics workspace table search-job	Run search jobs in Azure Monitor
az monitor log-profiles	Stream Azure Monitor activity log data
az monitor metrics	Monitor Azure Blob Storage
	Monitoring Azure Functions
	Monitor Azure Queue Storage
	Best practices for monitoring Azure Queue Storage
	Analyze Azure Files metrics using Azure Monitor
	Azure Monitor autoscaling common metrics
	Monitor container resources in Azure Container Instances
	Monitor Azure Table Storage
	Create metric alert in Azure CLI
	Get Load Balancer metrics with Azure Monitor CLI
az monitor metrics alert	Manage your alert rules

Reference subgroup	Azure CLI article showing reference use
	Release and work item insights
	Create a new alert rule using the CLI, PowerShell, or an ARM template
	Create metric alert in Azure CLI
	Set up metric alerts for Azure Stack HCI
az monitor metrics alert condition	Create metric alert in Azure CLI
az monitor metrics alert dimension	Create metric alert in Azure CLI
az monitor scheduled-query	Manage your alert rules
	Move a Log Analytics workspace to a different subscription or resource group
	Create a new alert rule using the CLI, PowerShell, or an ARM template

az mysql

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az mysql	Quickstart: Create an Azure Database for MySQL server using Azure CLI
	Deploy a Spring app to App Service with MySQL
	Quickstart: Create an Azure Database for MySQL using a simple Azure CLI command - az mysql up (preview)
az mysql db	Tutorial: Create a multi-container (preview) app in Web App for Containers
	Quickstart: Use Java and JDBC with Azure Database for MySQL
	Manage an Azure Database for MySQL single server using the Azure CLI
az mysql flexible-server	Major version upgrade in Azure Database for MySQL - Flexible Server
	Quickstart: Connect with Azure Database for MySQL - Flexible Server by using Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Connect an App Services Web app to Azure Database for MySQL - Flexible Server in a virtual network
	Tutorial: Deploy WordPress app on AKS with Azure Database for MySQL - Flexible Server
	Tutorial: Deploy a Spring application to Azure Spring Apps with a passwordless connection to an Azure database
	Quickstart: Create an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Example: Use the Azure libraries to create a database
	Use Java and JDBC with Azure Database for MySQL - Flexible Server
	Tutorial: Deploy a PHP and Azure Database for MySQL - Flexible Server app on Azure App Service
	Tutorial: Deploy a Spring Boot application on AKS cluster with Azure Database for MySQL - Flexible Server in a VNet
az mysql flexible-server ad-admin	Set up Microsoft Entra authentication for Azure Database for MySQL - Flexible Server
	Use Java and JDBC with Azure Database for MySQL - Flexible Server
	Migrate an application to use passwordless connections with Azure Database for MySQL
	Configure passwordless database connections for Java apps on Oracle WebLogic Servers
az mysql flexible-server db	Quickstart: Use Terraform to create an Azure Database for MySQL - Flexible Server instance
	Quickstart: Connect with Azure Database for MySQL - Flexible Server by using Azure CLI
	Tutorial: Deploy a Spring application to Azure Spring Apps with a passwordless connection to an Azure database
	Quickstart: Create an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Example: Use the Azure libraries to create a database
	Use Java and JDBC with Azure Database for MySQL - Flexible Server
	Tutorial: Deploy a PHP and Azure Database for MySQL - Flexible Server

Reference subgroup	Azure CLI article showing reference use
	app on Azure App Service
	Tutorial: Deploy a Spring Boot application on AKS cluster with Azure Database for MySQL - Flexible Server in a VNet
	Manage Azure Database for MySQL - Flexible Server using the Azure CLI
	Use Spring Data R2DBC with Azure Database for MySQL
az mysql flexible-server firewall-rule	Example: Use the Azure libraries to create a database
	Use Java and JDBC with Azure Database for MySQL - Flexible Server
	Tutorial: Deploy a PHP and Azure Database for MySQL - Flexible Server app on Azure App Service
	Manage firewall rules for Azure Database for MySQL - Flexible Server using Azure CLI
	Quickstart: Deploy JBoss EAP on Azure Red Hat OpenShift using the Azure portal
	Restart/stop/start an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Use Spring Data R2DBC with Azure Database for MySQL
	Configure passwordless database connections for Java apps on Oracle WebLogic Servers
	Create an Azure Database for MySQL - Flexible Server instance and enable public access connectivity using Azure CLI
	Tutorial: Connect to a MySQL Database from Java JBoss EAP App Service with passwordless connection
az mysql flexible-server gtid	How to configure Azure Database for MySQL - Flexible Server data-in replication
az mysql flexible-server identity	Use Java and JDBC with Azure Database for MySQL - Flexible Server
	Migrate an application to use passwordless connections with Azure Database for MySQL
	Configure passwordless database connections for Java apps on Oracle WebLogic Servers
az mysql flexible-server import	Migrate Azure Database for MySQL - Single Server to Flexible Server using Azure Database for MySQL Import CLI

Reference subgroup	Azure CLI article showing reference use
	Migrate MySQL on-premises or Virtual Machine (VM) workload to Azure Database for MySQL - Flexible Server using Azure Database for MySQL Import CLI (Public Preview)
az mysql flexible-server parameter	Tutorial: Query Performance Insight for Azure Database for MySQL - Flexible Server
	Tutorial: Configure audit logs by using Azure Database for MySQL - Flexible Server
	Configure server parameters in Azure Database for MySQL - Flexible Server using the Azure CLI
	List and change server parameters of Azure Database for MySQL - Flexible Server using Azure CLI
	Configure audit logs on Azure Database for MySQL - Flexible Server using Azure CLI
	Configure slow query logs on an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Tutorial: Install a LEMP stack on an Azure Linux VM
az mysql flexible-server replica	How to create and manage read replicas in Azure Database for MySQL - Flexible Server using the Azure CLI
	Create and manage read replicas in Azure Database for MySQL - Flexible Server using Azure CLI
az mysql flexible-server server-logs	Error logs in Azure Database for MySQL - Flexible Server (Preview)
	List and download Azure Database for MySQL - Flexible Server logs by using the Azure CLI
az mysql server	Tutorial: Create a multi-container (preview) app in Web App for Containers
	Configure SSL connectivity in your application to securely connect to Azure Database for MySQL
	Quickstart: Create an Azure Database for MySQL server using Azure CLI
	Major version upgrade in Azure Database for MySQL - Flexible Server
	Quickstart: Create an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Stop/Start an Azure Database for MySQL

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Use Java and JDBC with Azure Database for MySQL
	Major version upgrade in Azure Database for MySQL single server
	How to back up and restore a server in Azure Database for MySQL using the Azure CLI
	Create and manage Azure Database for MySQL firewall rules by using the Azure CLI
az mysql server add-admin	Quickstart: Use Java and JDBC with Azure Database for MySQL
az mysql server configuration	Tutorial: Query Performance Insight for Azure Database for MySQL - Flexible Server
	Deploy a Spring app to App Service with MySQL
	Configure server parameters in Azure Database for MySQL using the Azure CLI
	Configure and access audit logs in the Azure CLI
	Configure and access slow query logs by using Azure CLI
	List and update configurations of an Azure Database for MySQL server using Azure CLI
	Enable and download server slow query logs of an Azure Database for MySQL server using Azure CLI
az mysql server firewall-rule	Tutorial: Create a multi-container (preview) app in Web App for Containers
	Quickstart: Create an Azure Database for MySQL server using Azure CLI
	Quickstart: Use Java and JDBC with Azure Database for MySQL
	Migrate an application to use passwordless connections with Azure Database for MySQL
	Create and manage Azure Database for MySQL firewall rules by using the Azure CLI
	Moodle manual migration steps
	Tutorial: Design an Azure Database for MySQL using Azure CLI
	Create a MySQL server and configure a firewall rule using the Azure CLI
az mysql server key	Validating data encryption for Azure Database for MySQL

Reference subgroup	Azure CLI article showing reference use
	Data encryption for Azure Database for MySQL by using the Azure CLI
<code>az mysql server replica</code>	How to create and manage read replicas in Azure Database for MySQL using the Azure CLI and REST API
<code>az mysql server vnet-rule</code>	Create and manage Azure Database for MySQL VNet service endpoints using Azure CLI
<code>az mysql server-logs</code>	Configure and access slow query logs by using Azure CLI
	Enable and download server slow query logs of an Azure Database for MySQL server using Azure CLI

az netappfiles

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
<code>az netappfiles account</code>	Quickstart: Set up Azure NetApp Files and create an NFS volume
	Configure customer-managed keys for Azure NetApp Files volume encryption
	Configure Azure NetApp Files for Azure Kubernetes Service
<code>az netappfiles pool</code>	Quickstart: Set up Azure NetApp Files and create an NFS volume
	Configure Azure NetApp Files for Azure Kubernetes Service
<code>az netappfiles volume</code>	Quickstart: Set up Azure NetApp Files and create an NFS volume
	Provision Azure NetApp Files NFS volumes for Azure Kubernetes Service
	Provision Azure NetApp Files SMB volumes for Azure Kubernetes Service
	Provision Azure NetApp Files dual-protocol volumes for Azure Kubernetes Service
	Use Azure HPC Cache with Azure NetApp Files

az network

Reference subgroup	Azure CLI article showing reference use
az network	Check resource usage against limits
	Secure an Azure Machine Learning training environment with virtual networks
	Planning for migration of IaaS resources from classic to Azure Resource Manager
az network alb	Quickstart: Create Application Gateway for Containers - bring your own deployment
	Backend MTLS with Application Gateway for Containers - Gateway API
	SSL offloading with Application Gateway for Containers - Gateway API
	Path, header, and query string routing with Application Gateway for Containers - Gateway API
	Multi-site hosting with Application Gateway for Containers - Gateway API
	Multi-site hosting with Application Gateway for Containers - Ingress API
	Traffic splitting with Application Gateway for Containers - Gateway API
	URL Rewrite for Azure Application Gateway for Containers - Gateway API
	Header rewrite for Azure Application Gateway for Containers - Ingress API
	Header rewrite for Azure Application Gateway for Containers - Gateway API
az network alb association	Quickstart: Create Application Gateway for Containers - bring your own deployment
az network alb frontend	Quickstart: Create Application Gateway for Containers - bring your own deployment
	Backend MTLS with Application Gateway for Containers - Gateway API
az network application-gateway	Overview: Azure CLI terminology and support levels

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Enable application gateway ingress controller add-on for an existing AKS cluster with an existing application gateway
	Tutorial: Enable the ingress controller add-on for a new AKS cluster with a new application gateway instance
	Overview of mutual authentication with Application Gateway
	Install an Application Gateway Ingress Controller (AGIC) using an existing Application Gateway
	Application Gateway integration
	Expose a static IP address for a container group
	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
	Application Gateway - Backend health
	Quickstart: Direct web traffic with Azure Application Gateway - Azure CLI
az network application-gateway address-pool	Networking for Azure Virtual Machine Scale Sets
	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
	Create an application gateway that hosts multiple web sites using the Azure CLI
	Route web traffic based on the URL using the Azure CLI
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
az network application-gateway frontend-ip	Configure Azure Application Gateway Private Link
az network application-gateway frontend-port	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
	Route web traffic based on the URL using the Azure CLI
	Create an application gateway with HTTP to HTTPS redirection using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Create an application gateway with external redirection using the Azure CLI
<code>az network application-gateway http-listener</code>	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
	Create an application gateway that hosts multiple web sites using the Azure CLI
	Route web traffic based on the URL using the Azure CLI
	Create an application gateway with HTTP to HTTPS redirection using the Azure CLI
	Create an application gateway with external redirection using the Azure CLI
	Create an application gateway with internal redirection using the Azure CLI
<code>az network application-gateway http-settings</code>	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
	Expose applications to the internet with TLS Termination at Application Gateway
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
	Expose applications with end-to-end TLS in a virtual network
<code>az network application-gateway private-link</code>	Configure Azure Application Gateway Private Link
<code>az network application-gateway private-link ip-config</code>	Overview: Azure CLI terminology and support levels
<code>az network application-gateway probe</code>	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
<code>az network application-gateway redirect-config</code>	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Create an application gateway with HTTP to HTTPS redirection using the Azure CLI
	Create an application gateway with external redirection using the Azure CLI
	Create an application gateway with internal redirection using the Azure CLI
az network application-gateway rewrite-rule	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
az network application-gateway rewrite-rule condition	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
az network application-gateway rewrite-rule set	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
az network application-gateway root-cert	Expose applications with end-to-end TLS in a virtual network
az network application-gateway rule	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
	Create an application gateway that hosts multiple web sites using the Azure CLI
	Route web traffic based on the URL using the Azure CLI
	Create an application gateway with HTTP to HTTPS redirection using the Azure CLI
	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
	Create an application gateway with external redirection using the Azure CLI
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
	Create an application gateway with internal redirection using the Azure CLI
az network application-gateway ssl-cert	Renew Application Gateway certificates

Reference subgroup	Azure CLI article showing reference use
az network application-gateway url-path-map	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI Route web traffic based on the URL using the Azure CLI
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
az network application-gateway url-path-map rule	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI Route web traffic based on the URL using the Azure CLI
az network application-gateway waf-config	Customize Web Application Firewall rules using the Azure CLI Restrict web traffic using the Azure CLI
az network application-gateway waf-policy	Enable Web Application Firewall using the Azure CLI
az network application-gateway waf-policy custom-rule	Create rate limiting custom rules for Application Gateway WAF v2
az network application-gateway waf-policy custom-rule match-condition	Create rate limiting custom rules for Application Gateway WAF v2
az network application-gateway waf-policy managed-rule exclusion	Web Application Firewall exclusion lists
az network application-gateway waf-policy managed-rule exclusion rule-set	Web Application Firewall exclusion lists
az network application-gateway waf-policy policy-setting	How to mask sensitive data on Azure Web Application Firewall
az network asg	Create, change, or delete a network security group
	Configure an application security group with a private endpoint
	Filter network traffic with a network security group using the Azure CLI
az network bastion	Connect to a VM via specified private IP address
	File transfer using a native client
	Connect to a VM using Bastion and the Windows native

Reference subgroup	Azure CLI article showing reference use
	client
	Connect to a VM using Bastion and a Linux native client
	Quickstart: Use the Azure CLI to create a virtual network
	Quickstart: Create a private endpoint by using the Azure CLI
	Diagnose network security rules
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Set up Oracle ASM on an Azure Linux virtual machine
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
az network cross-region-lb	Tutorial: Create a cross-region Azure Load Balancer using Azure CLI
az network cross-region-lb ress-pool ress	Tutorial: Create a cross-region Azure Load Balancer using Azure CLI
az network cross-region-lb rule	Tutorial: Create a cross-region Azure Load Balancer using Azure CLI
az network custom-ip prefix	Create a custom IPv4 address prefix using the Azure CLI
	Create a custom IPv6 address prefix using Azure CLI
az network ddos-protection	Quickstart: Create and configure Azure DDoS Network Protection using Terraform
	QuickStart: Create and configure Azure DDoS Network Protection using Azure CLI
	Quickstart: Create and configure Azure DDoS Network Protection using ARM template
az network dns record-set	Import and export a DNS zone file using the Azure CLI
	Host reverse DNS lookup zones in Azure DNS
	Manage DNS records and recordsets in Azure DNS using the Azure CLI
	Quickstart: Create an Azure DNS zone and record using Azure CLI

Reference subgroup	Azure CLI article showing reference use
	How to manage DNS Zones in Azure DNS using the Azure CLI
	Azure CLI script example: Create a DNS zone and record
<code>az network dns record-set a</code>	Use TLS with an ingress controller on Azure Kubernetes Service (AKS)
	Manage DNS records and recordsets in Azure DNS using the Azure CLI
	Quickstart: Create an Azure DNS zone and record using Azure CLI
	Azure Front Door: Deploy custom domain
	Azure CLI script example: Create a DNS zone and record
<code>az network dns record-set aaaa</code>	Manage DNS records and recordsets in Azure DNS using the Azure CLI
<code>az network dns record-set caa</code>	Manage DNS records and recordsets in Azure DNS using the Azure CLI
<code>az network dns record-set cname</code>	Manage DNS records and recordsets in Azure DNS using the Azure CLI
	Azure Front Door: Deploy custom domain
	Reliability in Azure DNS
<code>az network dns record-set mx</code>	Manage DNS records and recordsets in Azure DNS using the Azure CLI
<code>az network dns record-set ns</code>	Import and export a DNS zone file using the Azure CLI
	Manage DNS records and recordsets in Azure DNS using the Azure CLI
	Quickstart: Create an Azure DNS zone and record using Azure CLI
<code>az network dns record-set ptr</code>	Host reverse DNS lookup zones in Azure DNS
	Manage DNS records and recordsets in Azure DNS using the Azure CLI
<code>az network dns record-set soa</code>	Manage DNS records and recordsets in Azure DNS using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
az network dns record-set srv	Manage DNS records and recordsets in Azure DNS using the Azure CLI
az network dns record-set txt	Manage DNS records and recordsets in Azure DNS using the Azure CLI
az network dns zone	<p>Overview: Azure CLI terminology and support levels</p> <p>Set up a custom domain name and SSL certificate with the application routing add-on</p>
	Import and export a DNS zone file using the Azure CLI
	Host reverse DNS lookup zones in Azure DNS
	Quickstart: Create an Azure private DNS zone using the Azure CLI
	Quickstart: Create an Azure DNS zone and record using Terraform
	Quickstart: Create an Azure DNS zone and record using Azure CLI
	Install Azure CLI on Azure Stack Hub
	How to manage DNS Zones in Azure DNS using the Azure CLI
	Configure NGINX ingress controller to support Azure private DNS zone with application routing add-on
az network express-route	Create and manage ExpressRoute public peering
	Quickstart: Create and modify an ExpressRoute circuit using Azure CLI
	Create and modify peering for an ExpressRoute circuit using CLI
	Connect a virtual network to an ExpressRoute circuit using Azure CLI
	Configure ExpressRoute Direct by using the Azure CLI
	Configure ExpressRoute Global Reach by using the Azure CLI
	Add IPv6 support for private peering using Azure CLI

Reference subgroup	Azure CLI article showing reference use
az network express-route auth	Connect a virtual network to an ExpressRoute circuit using Azure CLI
	Configure ExpressRoute Global Reach by using the Azure CLI
az network express-route peering	Create and manage ExpressRoute public peering
	Create and modify peering for an ExpressRoute circuit using CLI
	Configure route filters for Microsoft peering: Azure CLI
	Add IPv6 support for private peering using Azure CLI
az network express-route peering connection	Configure ExpressRoute Global Reach by using the Azure CLI
az network express-route port	Configure ExpressRoute Direct by using the Azure CLI
az network express-route port location	Configure ExpressRoute Direct by using the Azure CLI
az network firewall	Overview: Azure CLI terminology and support levels
	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Azure Firewall DNS settings
	Azure Firewall SNAT private IP address ranges
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters
	Configure a single public IP address for outbound and inbound traffic to a container group
	Quickstart: Deploy Azure Firewall with Availability Zones - Terraform
	Deploy and configure Azure Firewall using Azure CLI
	Control egress traffic for an Azure Spring Apps instance
	Use firewall to restrict outbound traffic using Azure CLI
az network firewall application-rule	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)

Reference subgroup	Azure CLI article showing reference use
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters
	Configure a single public IP address for outbound and inbound traffic to a container group
	Configure Azure Firewall application rules with SQL FQDNs
	Deploy and configure Azure Firewall using Azure CLI
	Control egress traffic for an Azure Spring Apps instance
	Use firewall to restrict outbound traffic using Azure CLI
	Restrict egress traffic from SQL Server 2019 Big Data Clusters in Azure Kubernetes Service (AKS) private cluster
az network firewall ip-config	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters
	Configure a single public IP address for outbound and inbound traffic to a container group
	Deploy and configure Azure Firewall using Azure CLI
	Control egress traffic for an Azure Spring Apps instance
	Use firewall to restrict outbound traffic using Azure CLI
	Restrict egress traffic from SQL Server 2019 Big Data Clusters in Azure Kubernetes Service (AKS) private cluster
az network firewall nat-rule	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters
	Configure a single public IP address for outbound and inbound traffic to a container group
az network firewall network-rule	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters

Reference subgroup	Azure CLI article showing reference use
	Deploy and configure Azure Firewall using Azure CLI
	Control egress traffic for an Azure Spring Apps instance
	Use firewall to restrict outbound traffic using Azure CLI
	Restrict egress traffic from SQL Server 2019 Big Data Clusters in Azure Kubernetes Service (AKS) private cluster
az network firewall policy	Secure traffic destined to private endpoints in Azure Virtual WAN
az network front-door	Configure an IP restriction rule with a WAF for Azure Front Door
	Tutorial: Quickly scale and protect a web application by using Azure Front Door and Azure Web Application Firewall (WAF)
	Azure Front Door: Deploy custom domain
	Quickstart: Create a Front Door for a highly available global web application using Azure CLI
az network front-door frontend-endpoint	Azure Front Door: Deploy custom domain
az network front-door routing-rule	Tutorial: Configure your rules engine
	Azure Front Door: Deploy custom domain
az network front-door rules-engine rule	Tutorial: Configure your rules engine
az network front-door rules-engine rule action	Tutorial: Configure your rules engine
az network front-door waf-policy	Configure an IP restriction rule with a WAF for Azure Front Door
	Configure a Web Application Firewall rate-limit rule
	Tutorial: Quickly scale and protect a web application by using Azure Front Door and Azure Web Application Firewall (WAF)
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
	Tutorial: Add and customize delivery rules for Azure Front

Reference subgroup	Azure CLI article showing reference use
	Door Standard/Premium (Preview) with Azure CLI
az network front-door waf-policy managed-rules	Configure bot protection for Web Application Firewall
	Tutorial: Quickly scale and protect a web application by using Azure Front Door and Azure Web Application Firewall (WAF)
	Quickstart: Create an Azure Front Door Standard/Premium - Azure CLI
az network front-door waf-policy managed-rules exclusion	Configure web application firewall exclusion lists
az network front-door waf-policy rule	Configure an IP restriction rule with a WAF for Azure Front Door
	Configure a Web Application Firewall rate-limit rule
az network front-door waf-policy rule match-condition	Configure an IP restriction rule with a WAF for Azure Front Door
	Configure a Web Application Firewall rate-limit rule
az network ip-group	Quickstart: Create an Azure Firewall and IP Groups - Terraform
	Quickstart: Create an Azure Firewall with multiple public IP addresses - Terraform
	Create IP Groups
az network lb	Monitoring load balancer
	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	Manage Upgrade Policies for Virtual Machine Scale Sets
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Tutorial: Connect to Virtual Machine Scale Set instances using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
	High-availability SAP NetWeaver with simple mount and NFS on SLES for SAP Applications VMs
<code>az network lb address-pool</code>	Backend pool management
	Deploy an IPv6 dual stack application in Azure virtual network using Azure CLI
	Load balancing on multiple IP configurations using Azure CLI
	Create a public load balancer with IPv6 using Azure CLI
	Deploy a dual-stack Azure Gateway Load Balancer
	Add IPv6 to an IPv4 application in Azure virtual network using Azure CLI
	Load balance multiple websites
	Deploy an IPv6 dual stack application using Basic Load Balancer - CLI
<code>az network lb frontend-ip</code>	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
	High-availability SAP NetWeaver with simple mount and NFS on SLES for SAP Applications VMs
	High availability for SAP HANA scale-out system with HSR on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on VMs on RHEL with NFS on Azure Files

Reference subgroup	Azure CLI article showing reference use
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with Azure NetApp Files for SAP applications
	Access an app in Azure Spring Apps in a virtual network
	Azure Virtual Machines high availability for SAP NetWeaver on Red Hat Enterprise Linux
az network lb inbound-nat-rule	Manage inbound NAT rules for Azure Load Balancer
	Configure inbound NAT Rules for Virtual Machine Scale Sets
	Create a public load balancer with IPv6 using Azure CLI
az network lb outbound-rule	Use a public standard load balancer in Azure Kubernetes Service (AKS)
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
az network lb probe	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	Manage Upgrade Policies for Virtual Machine Scale Sets
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
	High-availability SAP NetWeaver with simple mount and NFS on SLES for SAP Applications VMs
	High availability for SAP HANA scale-out system with HSR on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on VMs on RHEL with NFS on Azure Files
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with Azure NetApp Files for

Reference subgroup	Azure CLI article showing reference use
	SAP applications
az network lb ress-pool tunnel-interface	Tutorial: Create a gateway load balancer using the Azure CLI
az network lb rule	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	Configure TCP reset and idle timeout for Azure Load Balancer
	Manage Upgrade Policies for Virtual Machine Scale Sets
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	Configure the distribution mode for Azure Load Balancer
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Tutorial: Install applications in Virtual Machine Scale Sets with the Azure CLI
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
	High-availability SAP NetWeaver with simple mount and NFS on SLES for SAP Applications VMs
az network local-gateway	Create a virtual network with a site-to-site VPN connection using CLI
	How to configure BGP for Azure VPN Gateway: CLI
az network manager	Quickstart: Create a mesh network topology with Azure Virtual Network Manager using Terraform
	Configure a cross-tenant connection in Azure Virtual Network Manager Preview - CLI
	Quickstart: Create a mesh network topology with Azure Virtual Network Manager by using the Azure CLI
az network manager connect-config	Quickstart: Create a mesh network topology with Azure Virtual Network Manager by using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
az network manager connection subscription	Configure a cross-tenant connection in Azure Virtual Network Manager Preview - CLI
az network manager group	Quickstart: Create a mesh network topology with Azure Virtual Network Manager by using the Azure CLI
az network manager group static-member	Configure a cross-tenant connection in Azure Virtual Network Manager Preview - CLI
	Quickstart: Create a mesh network topology with Azure Virtual Network Manager by using the Azure CLI
az network manager scope-connection	Configure a cross-tenant connection in Azure Virtual Network Manager Preview - CLI
az network nat gateway	Azure NAT Gateway integration
	Create a managed or user-assigned NAT gateway for your Azure Kubernetes Service (AKS) cluster
	Scale SNAT ports with Azure NAT Gateway
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Manage NAT gateway
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
	Configure a NAT gateway for static IP address for outbound traffic from a container group
	Quickstart: Create an internal load balancer to load balance VMs using the Azure CLI
	Troubleshoot connections to endpoints outside the virtual network
	Quickstart: Create a NAT gateway using the Azure CLI
az network nic	Set shell variables from CLI output
	Create, change, or delete a network interface
	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Backend pool management

Reference subgroup	Azure CLI article showing reference use Configuring Azure Files network endpoints
	How to use your workspace with a custom DNS server
	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	Diagnose a virtual machine routing problem
	Use Azure CLI to create a Windows or Linux VM with Accelerated Networking
az network nic ip-config	Create, change, or delete a network interface
	Associate a public IP address to a virtual machine
	Configure IP addresses for an Azure network interface
	Dissociate a public IP address from an Azure VM
	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	How to create a Linux virtual machine in Azure with multiple network interface cards
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
az network nic ip-config inbound-nat-rule	Manage inbound NAT rules for Azure Load Balancer
az network nic ip-config ress-pool	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
	High-availability SAP NetWeaver with simple mount and NFS on SLES for SAP Applications VMs
	High availability for SAP HANA scale-out system with HSR on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on VMs on RHEL with NFS on Azure Files
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with Azure NetApp Files for SAP applications
az network nic vtap-config	Work with a virtual network TAP using the Azure CLI
az network nsg	Create, change, or delete a network security group
	Use Azure CLI to create a Windows or Linux VM with Accelerated Networking
	Create an Oracle Database in an Azure VM
	Test private endpoints by deploying Azure Load Testing in an Azure virtual network
	Resource logging for a network security group
	Diagnose network security rules
	How to create a Linux virtual machine in Azure with multiple network interface cards
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Set up Oracle ASM on an Azure Linux virtual machine
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
az network nsg rule	Create, change, or delete a network security group
	How to connect using Secure Shell (SSH) and sign on to an Azure virtual machine running Windows

Reference subgroup	Azure CLI article showing reference use
	Use Azure CLI to create a Windows or Linux VM with Accelerated Networking
	Create an Oracle Database in an Azure VM
	Use service tags with Power BI
	Diagnose network security rules
	How to create a Linux virtual machine in Azure with multiple network interface cards
	Quickstart: Create a public load balancer to load balance VMs using the Azure CLI
	Set up Oracle ASM on an Azure Linux virtual machine
	Tutorial: Configure dual stack outbound connectivity with a NAT gateway and a public load balancer
az network private-dns link vnet	Create a private Azure Kubernetes Service (AKS) cluster
	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Configure Azure Private Link for an Azure Cosmos DB account
	Configuring Azure Files network endpoints
	Provide a virtual network to an external Azure Container Apps environment
	Configure a private endpoint for an Azure Machine Learning workspace
	Provide a virtual network to an internal Azure Container Apps environment
	Secure an Azure Machine Learning training environment with virtual networks
	Quickstart: Create a private endpoint by using the Azure CLI
az network private-dns record-set	Troubleshooting online endpoints deployment and scoring
	Integrate Key Vault with Azure Private Link

Reference subgroup	Azure CLI article showing reference use Secure your managed online endpoints with network isolation
	Quickstart: Create an Azure private DNS zone using the Azure CLI
	Import and export a private DNS zone file for Azure private DNS
	Integrate Managed HSM with Azure Private Link
<code>az network private-dns record-set a</code>	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Configuring Azure Files network endpoints
	Provide a virtual network to an external Azure Container Apps environment
	Provide a virtual network to an internal Azure Container Apps environment
	Configuring Azure File Sync network endpoints
	Quickstart: Create an Azure private DNS zone using the Azure CLI
	Access an app in Azure Spring Apps in a virtual network
	Integrate Managed HSM with Azure Private Link
	Integrate Azure HPC Cache with Azure Kubernetes Service (AKS)
<code>az network private-dns zone</code>	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Configure Azure Private Link for an Azure Cosmos DB account
	Configuring Azure Files network endpoints
	Provide a virtual network to an external Azure Container Apps environment
	Configure a private endpoint for an Azure Machine Learning workspace

Reference subgroup	Azure CLI article showing reference use
	Provide a virtual network to an internal Azure Container Apps environment
	Quickstart: Create a private endpoint by using the Azure CLI
	Migrate Azure Database for MySQL - Single Server to Flexible Server using Azure Database for MySQL Import CLI
	Configuring Azure File Sync network endpoints
az network private-endpoint	Use an internal load balancer with Azure Kubernetes Service (AKS)
	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Configure Azure Private Link for an Azure Cosmos DB account
	Manage Azure private endpoints
	Configuring Azure Files network endpoints
	Configure a private endpoint for an Azure Machine Learning workspace
	What is Azure Cache for Redis with Azure Private Link?
	How to use your workspace with a custom DNS server
	Set up Private Link with Azure Virtual Desktop
az network private-endpoint asg	Configure an application security group with a private endpoint
az network private-endpoint dns-zone-group	Configure Azure Private Link for an Azure Cosmos DB account
	Configure a private endpoint for an Azure Machine Learning workspace
	Quickstart: Create a private endpoint by using the Azure CLI
	Tutorial: Create a secure n-tier app in Azure App Service

Reference subgroup	Azure CLI article showing reference use
	Use Azure Private Link in Azure Cosmos DB for MongoDB vCore
	How to configure a private link for Azure AI
	Tutorial: Isolate back-end communication in Azure App Service with Virtual Network integration
	Tutorial: Connect to an Azure SQL server using an Azure Private Endpoint using Azure CLI
	Manage Azure Machine Learning workspaces using Azure CLI
	Manage your Azure AI Search service with the Azure CLI
az network private-endpoint-connection	Troubleshooting online endpoints deployment and scoring
	Manage Azure private endpoints
	Secure your managed online endpoints with network isolation
	Set up private access in Azure App Configuration
	Set up private access (preview)
	Access Key Vault in a private network through shared private endpoints
	Manage your Azure AI Search service with the Azure CLI
	Configure network access for Azure Elastic SAN
	Access Key Vault in private network through shared private endpoints
	Secure Azure SignalR outbound traffic through shared private endpoints
az network private-link-resource	Manage Azure private endpoints
	Use Azure Private Link in Azure Cosmos DB for MongoDB vCore
	Create and manage Private Link for Azure Database for MySQL - Flexible Server using Azure CLI
az network private-link-service	Use an internal load balancer with Azure Kubernetes

Reference subgroup	Azure CLI article showing reference use
	Service (AKS)
	Quickstart: Create a Private Link service using Azure CLI
az network profile	Deploy container instances into an Azure virtual network
	Create and configure an Azure Kubernetes Services (AKS) cluster to use virtual nodes using Azure CLI
	Failed to delete a virtual network or subnet used by Azure Container Instances
az network public-ip	Set shell variables from CLI output
	Use a public standard load balancer in Azure Kubernetes Service (AKS)
	Tutorial: Enable application gateway ingress controller add-on for an existing AKS cluster with an existing application gateway
	Create, change, or delete a network interface
	Associate a public IP address to a virtual machine
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Use a static public IP address and DNS label with the Azure Kubernetes Service (AKS) load balancer
	Use TLS with an ingress controller on Azure Kubernetes Service (AKS)
	Advanced NGINX ingress controller and ingress configurations with the application routing add-on
az network public-ip prefix	Use a public standard load balancer in Azure Kubernetes Service (AKS)
	Use instance-level public IPs in Azure Kubernetes Service (AKS)
	Manage NAT gateway
	Quickstart: Create a public IP address prefix using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
az network route-filter	Configure route filters for Microsoft peering: Azure CLI
az network route-filter rule	Configure route filters for Microsoft peering: Azure CLI
az network routeserver	Quickstart: Create and configure Route Server using Azure CLI
	Configure routing preference to influence route selection using the Azure CLI
az network routeserver peering	Quickstart: Create and configure Route Server using Azure CLI
az network route-table	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters
	Test private endpoints by deploying Azure Load Testing in an Azure virtual network
	Configure a single public IP address for outbound and inbound traffic to a container group
	Deploy and configure Azure Firewall using Azure CLI
	Plan a virtual network for Azure HDInsight
	Virtual network integration for Azure Data Lake Storage Gen1
	Route network traffic with a route table using the Azure CLI
	Control egress traffic for an Azure Spring Apps instance
	Create an Azure SQL Managed Instance using the Azure CLI
az network route-table route	Virtual network traffic routing
	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Configure inbound and outbound network traffic
	Use Azure Firewall to protect Azure Kubernetes Service (AKS) clusters

Reference subgroup	Azure CLI article showing reference use
	Configure a single public IP address for outbound and inbound traffic to a container group
	Deploy and configure Azure Firewall using Azure CLI
	App Service Environment management addresses
	Virtual network integration for Azure Data Lake Storage Gen1
	Route network traffic with a route table using the Azure CLI
	Control egress traffic for an Azure Spring Apps instance
az network service-endpoint policy	Manage data exfiltration to Azure Storage accounts with virtual network service endpoint policies using the Azure CLI
	Configure virtual network service endpoint policies for Azure HDInsight
az network service-endpoint policy-definition	Manage data exfiltration to Azure Storage accounts with virtual network service endpoint policies using the Azure CLI
	Configure virtual network service endpoint policies for Azure HDInsight
az network traffic-manager endpoint	Quickstart: Create a Traffic Manager profile using Bicep
	Quickstart: Create a Traffic Manager profile for a highly available web application using Azure CLI
	Run an N-tier application in multiple Azure Stack Hub regions for high availability
	Traffic Manager subnet override using Azure CLI
	Route traffic for high availability of applications using Azure CLI
	Route traffic for high availability of applications - Azure CLI
az network traffic-manager profile	Quickstart: Create an Azure Traffic Manager profile using Terraform
	Quickstart: Create a Traffic Manager profile using Bicep

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create a Traffic Manager profile for a highly available web application using Azure CLI
	Route traffic for high availability of applications using Azure CLI
	Route traffic for high availability of applications - Azure CLI
az network vhub	Overview: Azure CLI terminology and support levels
az network vhub connection	Connect cross-tenant virtual networks to a Virtual WAN hub - with Azure CLI
az network virtual-appliance	About NVAs in a Virtual WAN hub
az network vnet	Create a virtual network
	Tutorial: Enable application gateway ingress controller add-on for an existing AKS cluster with an existing application gateway
	Deploy container instances into an Azure virtual network
	Create a virtual network peering - Resource Manager, different subscriptions and Microsoft Entra tenants
	Use kubenet networking with your own IP address ranges in Azure Kubernetes Service (AKS)
	Integrate Key Vault with Azure Private Link
	Configure Azure Private Link for an Azure Cosmos DB account
	Limit network traffic with Azure Firewall in Azure Kubernetes Service (AKS)
	Create, change, or delete a virtual network
	Configuring Azure Files network endpoints
az network vnet peering	Create a private Azure Kubernetes Service (AKS) cluster
	Tutorial: Enable application gateway ingress controller add-on for an existing AKS cluster with an existing application gateway
	Create a virtual network peering - Resource Manager, different subscriptions and Microsoft Entra tenants

Reference subgroup	Azure CLI article showing reference use
	Create, change, or delete a virtual network peering
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Create a virtual network peering - different deployment models and subscriptions
	Create a virtual network peering - different deployment models, same subscription
	Connect virtual networks with virtual network peering using the Azure CLI
	Quickstart: Create a multi-region cluster with Azure Managed Instance for Apache Cassandra
	Tutorial: Clean up resources
az network vnet subnet	Configure Azure Storage firewalls and virtual networks
	Configure Azure AI services virtual networks
	Use an internal load balancer with Azure Kubernetes Service (AKS)
	Create, change, or delete a network security group
	Manage network policies for private endpoints
	Use kubenet networking with your own IP address ranges in Azure Kubernetes Service (AKS)
	Add, change, or delete a virtual network subnet
	Connect privately to an Azure container registry using Azure Private Link
	Integrate Key Vault with Azure Private Link
	Configure Azure Private Link for an Azure Cosmos DB account
az network vnet tap	Work with a virtual network TAP using the Azure CLI
az network vnet-gateway	Reset a VPN gateway or a connection
	Create a route-based VPN gateway using CLI
	Configure a Point-to-Site (P2S) VPN on Linux for use with Azure Files

Reference subgroup	Azure CLI article showing reference use
	Create a virtual network with a site-to-site VPN connection using CLI
	Configure a VNet-to-VNet VPN gateway connection using Azure CLI
	How to configure BGP for Azure VPN Gateway: CLI
	Add IPv6 support for private peering using Azure CLI
<code>az network vnet-gateway root-cert</code>	Configure a Point-to-Site (P2S) VPN on Linux for use with Azure Files
<code>az network vnet-gateway vpn-client</code>	Configure a Point-to-Site (P2S) VPN on Linux for use with Azure Files
<code>az network vpn-connection</code>	Create a virtual network with a site-to-site VPN connection using CLI
	Configure a VNet-to-VNet VPN gateway connection using Azure CLI
	How to configure BGP for Azure VPN Gateway: CLI
	Connect a virtual network to an ExpressRoute circuit using Azure CLI
<code>az network vpn-connection shared-key</code>	Create a virtual network with a site-to-site VPN connection using CLI
<code>az network watcher</code>	View the topology of an Azure virtual network
	Enable or disable Azure Network Watcher
	Diagnose network security rules
	Troubleshoot connections with Azure Network Watcher using the Azure CLI
	Quickstart: Diagnose a virtual machine network traffic filter problem using the Azure CLI
	Diagnose a virtual machine network routing problem - Azure CLI
	Analyze your Virtual Machine security with Security Group View using Azure CLI
<code>az network watcher flow-log</code>	Create, change, enable, disable, or delete VNet flow logs using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Manage NSG flow logs using the Azure CLI
az network watcher packet-capture	Manage packet captures for virtual machines with Azure Network Watcher using the Azure CLI
az network watcher troubleshooting	Troubleshoot VPN virtual network gateways and connections using the Azure CLI

az network-analytics

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az network-analytics data-product	Create an Azure Operator Insights Data Product

az networkcloud

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az networkcloud baremetalmachine	BareMetal Machine roles Manage lifecycle of Bare Metal Machines
	Working with placement hints in Azure Operator Nexus virtual machine
	Introduction to the Microsoft Defender for Endpoint runtime protection service
	Troubleshoot bare metal machine issues using the az networkcloud baremetalmachine run-data-extract command
	CSN storage pod container stuck in ContainerCreating
	Troubleshoot BMM issues using the az networkcloud baremetalmachine run-read-command
	Enable node down cleaner
az networkcloud cloudservicesnetwork	Prerequisites for deploying tenant workloads

Reference subgroup	Azure CLI article showing reference use
	Complete the prerequisites to deploy Azure Operator 5G Core Preview on Nexus Azure Kubernetes Service
<code>az networkcloud cluster</code>	Prerequisites for deploying tenant workloads
	Create and provision a Cluster using Azure CLI
	Upgrading cluster runtime from Azure CLI
	Working with placement hints in Azure Operator Nexus virtual machine
	Introduction to the Microsoft Defender for Endpoint runtime protection service
	Service principal rotation on the target cluster
<code>az networkcloud cluster baremetalmachinekeyset</code>	Manage emergency access to a bare metal machine using the az networkcloud cluster baremetalmachinekeyset
<code>az networkcloud cluster bmckeyset</code>	Manage emergency access to a bare metal machine using the az networkcloud cluster bmckeyset
<code>az networkcloud cluster metricsconfiguration</code>	Cluster metrics configuration
<code>az networkcloud clustermanager</code>	Cluster Manager: How to manage the Cluster Manager in Operator Nexus
<code>az networkcloud kubernetescluster</code>	Quickstart: Deploy an Azure Nexus Kubernetes cluster using Bicep
	Quickstart: Create an Azure Nexus Kubernetes cluster by using Azure CLI
	Monitor Nexus Kubernetes cluster
	Connect to Azure Operator Nexus Kubernetes cluster
	Upgrade an Azure Operator Nexus Kubernetes cluster
	Quickstart: Deploy an Azure Nexus Kubernetes cluster by using Azure Resource Manager template (ARM template)
	Restart Azure Operator Nexus Kubernetes cluster node
	Role-based access control in Azure Operator Nexus Kubernetes clusters
	Complete the prerequisites to deploy Azure Operator 5G

Reference subgroup	Azure CLI article showing reference use
	Core Preview on Nexus Azure Kubernetes Service
az networkcloud kubernetescluster agentpool	Quickstart: Deploy an Azure Nexus Kubernetes cluster using Bicep
	Quickstart: Create an Azure Nexus Kubernetes cluster by using Azure CLI
	Quickstart: Deploy an Azure Nexus Kubernetes cluster by using Azure Resource Manager template (ARM template)
az networkcloud l2network	Prerequisites for deploying tenant workloads
az networkcloud l3network	Prerequisites for deploying tenant workloads
	Complete the prerequisites to deploy Azure Operator 5G Core Preview on Nexus Azure Kubernetes Service
az networkcloud rack	Working with placement hints in Azure Operator Nexus virtual machine
az networkcloud trunkednetwork	Prerequisites for deploying tenant workloads
	Complete the prerequisites to deploy Azure Operator 5G Core Preview on Nexus Azure Kubernetes Service
az networkcloud virtualmachine	Quickstart: Create an Azure Operator Nexus virtual machine by using Azure CLI
	Working with placement hints in Azure Operator Nexus virtual machine
	Quickstart: Create an Azure Operator Nexus virtual machine by using Bicep
	Quickstart: Create an Azure Operator Nexus virtual machine by using Azure Resource Manager template (ARM template)
az networkcloud virtualmachine console	Introduction to the Virtual Machine console service

az networkfabric

[] Expand table

Reference subgroup	Azure CLI article showing reference use
az networkfabric acl	Access Control List Creation and Configuration Examples

Reference subgroup	Azure CLI article showing reference use
	How-To Guide: Creating ACLs on an NNI
az networkfabric controller	Create and modify a Network Fabric Controller using Azure CLI Network Fabric Controller overview
az networkfabric device	Create and Provision a Network Fabric using Azure CLI
az networkfabric externalnetwork	Access Control List Creation and Configuration Examples Configuration examples for creating an isolation domain
	Configure L2 and L3 isolation-domains using managed network fabric services
az networkfabric fabric	Configuration options for PE-CE connectivity Access Control List Creation and Configuration Examples
	Create and Provision a Network Fabric using Azure CLI
az networkfabric internalnetwork	Configuration examples for creating an isolation domain Prerequisites for deploying tenant workloads
	Configure L2 and L3 isolation-domains using managed network fabric services
	Complete the prerequisites to deploy Azure Operator 5G Core Preview on Nexus Azure Kubernetes Service
az networkfabric ipcommunity	Route Policy in Network Fabric
az networkfabric ipextendedcommunity	Route Policy in Network Fabric
az networkfabric ipprefix	Route Policy in Network Fabric
az networkfabric l2domain	Configuration examples for creating an isolation domain Prerequisites for deploying tenant workloads
	Configure L2 and L3 isolation-domains using managed network fabric services
az networkfabric l3domain	Configuration examples for creating an isolation domain Prerequisites for deploying tenant workloads

Reference subgroup	Azure CLI article showing reference use
	Configure L2 and L3 isolation-domains using managed network fabric services
	Complete the prerequisites to deploy Azure Operator 5G Core Preview on Nexus Azure Kubernetes Service
az networkfabric neighborgroup	Network Packet Broker
	Neighbor Group Configuration Overview
az networkfabric nni	Access Control List Creation and Configuration Examples
	Create and Provision a Network Fabric using Azure CLI
	How-To Guide: Creating ACLs on an NNI
az networkfabric npb	Network Packet Broker
az networkfabric routepolicy	Route Policy in Network Fabric
az networkfabric tap	Network Packet Broker
az networkfabric taprule	Network Packet Broker

az notification-hub

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az notification-hub	Quickstart: Set up push notifications in a notification hub
	Use Private Link (preview)
	Quickstart: Create a notification hub using a Resource Manager template
	Quickstart: Create an Azure notification hub using the Azure CLI
	Quickstart: Create an Azure notification hub using Terraform
az notification-hub authorization-rule	Quickstart: Set up push notifications in a notification hub
	Quickstart: Create an Azure notification hub using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
az notification-hub credential gcm	Quickstart: Set up push notifications in a notification hub
az notification-hub namespace	Use Private Link (preview)
	Quickstart: Create a notification hub using a Resource Manager template
	Quickstart: Create an Azure notification hub using the Azure CLI

az peering

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az peering service	Create, change, or delete a Peering Service connection using the Azure CLI
az peering service country	Create, change, or delete a Peering Service connection using the Azure CLI
az peering service location	Create, change, or delete a Peering Service connection using the Azure CLI
az peering service prefix	Create, change, or delete a Peering Service connection using the Azure CLI
az peering service provider	Create, change, or delete a Peering Service connection using the Azure CLI

az pipelines

[] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az pipelines	Create your first pipeline
	Use a variable group's secret and nonsecret variables in an Azure Pipeline
	Manage your pipeline with Azure CLI

Reference subgroup	Azure CLI article showing reference use
az pipelines agent	Overview: Azure CLI terminology and support levels Azure Pipelines agents
az pipelines build	Azure DevOps CLI in Azure Pipeline YAML
az pipelines pool	Azure Pipelines agents Create and manage agent pools
az pipelines queue	Create and manage agent pools
az pipelines runs	Pipeline run sequence
az pipelines runs artifact	Publish and download pipeline artifacts Publish Pipeline Artifacts
az pipelines runs tag	Pipeline run sequence
az pipelines variable	Define variables
az pipelines variable-group	Add & use variable groups Task types & usage Use a variable group's secret and nonsecret variables in an Azure Pipeline
	Azure DevOps CLI in Azure Pipeline YAML Build a data pipeline by using Azure Data Factory, DevOps, and machine learning
az pipelines variable-group variable	Use SAP Deployment Automation Framework from Azure DevOps Services
	Add & use variable groups Use a variable group's secret and nonsecret variables in an Azure Pipeline
	Azure DevOps CLI in Azure Pipeline YAML

az policy

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az policy assignment	Enable monitoring for Kubernetes clusters Remediate non-compliant resources with Azure Policy Resolve errors for request disallowed by policy Built-in policies for Azure Monitor Quickstart: Create a policy assignment to identify non-compliant resources using Terraform Export Azure Policy resources Programmatically create policies Quickstart: Create a policy assignment to identify non-compliant resources using Azure CLI Connect Azure Arc-enabled servers to Microsoft Defender for Cloud Quickstart: Create a policy assignment to identify non-compliant resources by using a Bicep file
az policy definition	Enable monitoring for Kubernetes clusters Tutorial: Create and manage policies to enforce compliance Remediate non-compliant resources with Azure Policy Subscription deployments with ARM templates Resolve errors for request disallowed by policy Create diagnostic settings at scale using Azure Policies and Initiatives Export Azure Policy resources Programmatically create policies Quickstart: Create a policy assignment to identify non-compliant resources using Azure CLI Quickstart: Create a policy assignment to identify non-compliant resources by using a Bicep file
az policy identity	Remediate non-compliant resources with Azure Policy

Reference subgroup	Azure CLI article showing reference use
	Built-in policies for Azure Monitor
az policy remediation	Remediate non-compliant resources with Azure Policy
	Built-in policies for Azure Monitor
	Use Image Integrity to validate signed images before deploying them to your Azure Kubernetes Service (AKS) clusters (Preview)
az policy set-definition	Tutorial: Create and manage policies to enforce compliance
	Built-in policies for Azure Monitor
	Export Azure Policy resources
az policy state	Get compliance data of Azure resources
	Built-in policies for Azure Monitor
	Quickstart: Create a policy assignment to identify non-compliant resources using Azure CLI
	Quickstart: Create a policy assignment to identify non-compliant resources by using a Bicep file
	Quickstart: Create a policy assignment to identify non-compliant resources by using ARM template
	Audit compliance of Azure container registries using Azure Policy
	Audit compliance of Azure Web PubSub Service resources using Azure Policy
	Audit compliance of Azure SignalR Service resources using Azure Policy

az portal

Expand table

Reference subgroup	Azure CLI article showing reference use
az portal dashboard	Quickstart: Create an Azure portal dashboard with Azure CLI

az postgres

Reference subgroup	Azure CLI article showing reference use
az postgres	Quickstart: Create an Azure Database for PostgreSQL server by using the Azure CLI
	Tutorial: Using Service Connector to build a Django app with Postgres on Azure App Service
	Quickstart: Use the az postgres up command to create an Azure Database for PostgreSQL - Single Server
az postgres db	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL
	Tutorial: Connect to a PostgreSQL Database from Java Tomcat App Service without secrets using a managed identity
	Tutorial: Connect to PostgreSQL Database from a Java Quarkus Container App without secrets using a managed identity
	Manage an Azure Database for PostgreSQL Single server using the Azure CLI
	Deploy a Java application with Quarkus on an Azure Kubernetes Service cluster
	Deploy a Java application with Quarkus on an Azure Container Apps
	Tutorial: Install Red Hat JBoss EAP on Azure Virtual Machines manually
az postgres flexible-server	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Tutorial: Deploy Django app with App Service and Azure Database for PostgreSQL - Flexible Server
	Quickstart: Connect and query with Azure CLI with Azure Database for PostgreSQL - Flexible Server
	Build and deploy a Python web app with Azure Container Apps and PostgreSQL
	Create and manage read replicas in Azure Database for PostgreSQL - Flexible Server from the Azure portal, CLI or REST API
	Azure Pipelines task - Azure Database for PostgreSQL - Flexible Server
	Create and deploy a Flask Python web app to Azure with system-assigned managed identity

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Create a passwordless connection to a database service via Service Connector
	Quickstart: Create an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
	Quickstart: Deploy your first web application to Azure Spring Apps
az postgres flexible-server ad-admin	Create and deploy a Django web app to Azure with a user-assigned managed identity
az postgres flexible-server db	Deploy a PostgreSQL Flexible Server Database using Terraform
	Build and deploy a Python web app with Azure Container Apps and PostgreSQL
	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL - Flexible Server
	Tutorial: Connect to a PostgreSQL Database from Java Tomcat App Service without secrets using a managed identity
	Tutorial: Deploy a Spring application to Azure Spring Apps with a passwordless connection to an Azure database
	Tutorial: Connect to PostgreSQL Database from a Java Quarkus Container App without secrets using a managed identity
	Use Spring Data R2DBC with Azure Database for PostgreSQL
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
	Configure passwordless database connections for Java apps on Oracle WebLogic Servers
az postgres flexible-server firewall-rule	Build and deploy a Python web app with Azure Container Apps and PostgreSQL
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL - Flexible Server
	Create and deploy a Django web app to Azure with a user-assigned managed identity

Reference subgroup	Azure CLI article showing reference use
	Create and manage Azure Database for PostgreSQL - Flexible Server firewall rules using the Azure CLI
	Use Spring Data R2DBC with Azure Database for PostgreSQL
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
az postgres flexible-server migration	Tutorial: Migrate Azure Database for PostgreSQL - Single Server to Flexible Server by using the Azure CLI
	Tutorial: Offline migration to Azure Database for PostgreSQL from on-premises or Azure VM-hosted PostgreSQL using migration service Preview
	Premigration validations for the migrations service in Azure Database for PostgreSQL Preview
	Tutorial: Offline migration to Azure Database for PostgreSQL from AWS RDS PostgreSQL using migration service Preview
az postgres flexible-server parameter	PostgreSQL extensions in Azure Database for PostgreSQL - Flexible Server
	Audit logging in Azure Database for PostgreSQL - Flexible Server
	Customize server parameters for Azure Database for PostgreSQL - Flexible Server using Azure CLI
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
	Configure intelligent tuning for Azure Database for PostgreSQL - Flexible Server by using the Azure CLI
az postgres flexible-server replica	Create and manage read replicas in Azure Database for PostgreSQL - Flexible Server from the Azure portal, CLI or REST API
az postgres flexible-server server-logs	List and download Azure Database for PostgreSQL flexible server logs by using the Azure CLI
az postgres flexible-server virtual-endpoint	Create and manage read replicas in Azure Database for PostgreSQL - Flexible Server from the Azure portal, CLI or REST API
az postgres server	Configure TLS connectivity in Azure Database for PostgreSQL - Single Server
	Quickstart: Create an Azure Database for PostgreSQL server by using the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL
	Tutorial: Connect to a PostgreSQL Database from Java Tomcat App Service without secrets using a managed identity
	Tutorial: Connect to PostgreSQL Database from a Java Quarkus Container App without secrets using a managed identity
	Logical decoding
	How to back up and restore a server in Azure Database for PostgreSQL - Single Server using the Azure CLI
	Tutorial: Design an Azure Database for PostgreSQL - Single Server using Azure CLI
	Manage an Azure Database for PostgreSQL Single server using the Azure CLI
	Create and manage Private Link for Azure Database for PostgreSQL - Single server using CLI
az postgres server add-admin	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL
az postgres server configuration	Monitor performance with the Query Store
	Logical decoding
	Customize server configuration parameters for Azure Database for PostgreSQL - Single Server using Azure CLI
	List and update configurations of an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
	Create and manage read replicas from the Azure CLI, REST API
	Enable and download server slow query logs of an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
az postgres server firewall-rule	Quickstart: Create an Azure Database for PostgreSQL server by using the Azure CLI
	Quickstart: Use Java and JDBC with Azure Database for PostgreSQL
	Logical decoding
	Tutorial: Design an Azure Database for PostgreSQL - Single Server using Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Install Red Hat JBoss EAP on Azure Virtual Machines manually
	Create an Azure Database for PostgreSQL - Flexible Server instance and configure a firewall rule using the Azure CLI
az postgres server key	Validating data encryption for Azure Database for PostgreSQL
	Data encryption for Azure Database for PostgreSQL Single server by using the Azure CLI
az postgres server replica	Create and manage read replicas from the Azure CLI, REST API
az postgres server vnet-rule	Create and manage VNet service endpoints for Azure Database for PostgreSQL - Single Server using Azure CLI
	Create an Azure Database for PostgreSQL - Flexible Server instance and configure a vNet rule using the Azure CLI
az postgres server-arc	Storage Configuration
	Scale up and down an Azure Database for PostgreSQL server using CLI (az or kubectl)
	Create an Azure Arc-enabled PostgreSQL server from CLI
	Use PostgreSQL extensions in your Azure Arc-enabled PostgreSQL server
	Change the port on which the server group is listening
	Automated backup Azure Arc-enabled PostgreSQL servers
	Show the configuration of an Azure Arc-enabled PostgreSQL server
	Delete an Azure Arc-enabled PostgreSQL server
	Troubleshooting PostgreSQL servers
	Deploy Active Directory integrated Azure Arc-enabled PostgreSQL using Azure CLI
az postgres server-arc endpoint	Get connection endpoints & create the connection strings for your Azure Arc-enabled PostgreSQL server
	Create an Azure Arc-enabled PostgreSQL server from CLI
	View logs and metrics using Kibana and Grafana
	Migrate PostgreSQL database to Azure Arc-enabled PostgreSQL server

Reference subgroup	Azure CLI article showing reference use
az postgres server-logs	Configure and access server logs by using Azure CLI
	Enable and download server slow query logs of an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI

az powerbi

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az powerbi embedded-capacity	Create Power BI Embedded capacity in the Azure portal

az ppg

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az ppg	Use proximity placement groups to reduce latency for Azure Kubernetes Service (AKS) clusters
	Deploy VMs to proximity placement groups using Azure CLI

az private-link

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az private-link association	Use APIs to create a private link for managing Azure resources
	Manage resource management private links

az provider

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az provider	Move Azure resources to a new resource group or subscription Azure resource providers and types
	Azure Policy definition structure
	Quickstart: Deploy your first application to Azure Spring Apps
	Resolve errors for resource provider registration
	Connected Machine agent prerequisites
	Prerequisites for Azure Virtual Desktop
	Use a public standard load balancer in Azure Kubernetes Service (AKS)
	Configure Azure CNI Overlay networking in Azure Kubernetes Service (AKS)
	Log Analytics workspace data export in Azure Monitor
az provider operation	Azure Container Registry roles and permissions Resource functions for ARM templates
	Resource functions for Bicep
	Manage access to Azure Machine Learning workspaces
	Authorizing access to Event Grid resources

az purview

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az purview account	Tutorial: Set up a data product batch

az quantum

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az quantum job	Integrated hybrid computing ↗
az quantum offerings	Manage quantum workspaces with the Azure CLI ↗
az quantum workspace	Integrated hybrid computing ↗
	Azure Quantum Credits ↗
	Manage quantum workspaces with the Azure CLI ↗
	Azure Quantum quotas ↗

az redis

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az redis	Scale an Azure Cache for Redis instance
	Configure data persistence for an Azure Cache for Redis instance
	Managed identity for storage
	How to upgrade an existing Redis 4 cache to Redis 6
	Tutorial: Connect services in Azure Container Apps (preview)
	Create an Azure Cache for Redis using the Azure CLI
	Create a Premium Azure Cache for Redis with clustering
	Quickstart: Load application secrets using Key Vault
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
	Quickstart: Route Azure Cache for Redis events to web endpoint with Azure CLI
az redis identity	Managed identity for storage

az redisenterprise

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az redisenterprise	Scale an Azure Cache for Redis instance
	Configure data persistence for an Azure Cache for Redis instance
	Configure active geo-replication for Enterprise Azure Cache for Redis instances
az redisenterprise database	Configure data persistence for an Azure Cache for Redis instance
	Configure active geo-replication for Enterprise Azure Cache for Redis instances

az repos

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az repos	Branch policies and settings
	Set Git repository settings and policies
	Get started with Git from the command line
	Delete a Git repo from your project
	Auto detect configuration and git aliases
az repos import	Import a Git repo
az repos policy	Branch policies and settings
	Set Git repository settings and policies
az repos policy approver-count	Branch policies and settings
az repos policy build	Branch policies and settings
az repos policy case-enforcement	Set Git repository settings and policies
az repos policy comment-required	Branch policies and settings
az repos policy file-size	Set Git repository settings and policies
az repos policy merge-strategy	Branch policies and settings

Reference subgroup	Azure CLI article showing reference use
az repos policy required-reviewer	Branch policies and settings
az repos policy work-item-linking	Branch policies and settings
az repos pr	Create pull requests
	Complete, abandon, or revert pull requests
	Review pull requests
	View, filter, and open pull requests
	Auto detect configuration and git aliases
az repos pr policy	Complete, abandon, or revert pull requests
az repos pr reviewer	Create pull requests
az repos pr work-item	Create pull requests

az resource

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az resource	Learn Azure CLI syntax differences in Bash, PowerShell and Cmd
	Configure an App Service app
	Move Azure resources to a new resource group or subscription
	Azure Functions networking options
	Set up Azure App Service access restrictions
	Configure Azure AI services virtual networks
	Tutorial: Host a RESTful API with CORS in Azure App Service
	Configure a custom container for Azure App Service
	What are the resource providers for Azure services
	Troubleshoot Azure Kubernetes Service cluster/node in a failed state

az resourcemanagement

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az resourcemanagement private-link	Use APIs to create a private link for managing Azure resources
Manage resource management private links	

az resource-mover

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az resource-mover move-collection	Move a virtual machine in an availability zone using Azure PowerShell and CLI
az resource-mover move-resource	Move a virtual machine in an availability zone using Azure PowerShell and CLI

az rest

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Tips for using the Azure CLI successfully
	Learn Azure CLI syntax differences in Bash, PowerShell and Cmd
	Considerations for running the Azure CLI in a PowerShell environment
	Use the Azure REST API with Azure CLI
	What are availability zones?
	Elevate access to manage all Azure subscriptions and management groups
	Connect to Azure by using an Azure Resource Manager service connection
	Manage your function app
	High availability for Azure SQL Database

Reference subgroup	Azure CLI article showing reference use
	Migrate an API Management instance hosted on the stv1 platform to stv2

az role

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az role assignment	Create an Azure service principal with Azure CLI Manage service principal roles Provide access to Key Vault keys, certificates, and secrets with an Azure role-based access control Elevate access to manage all Azure subscriptions and management groups Assign an Azure role for access to blob data Log in to a Windows virtual machine in Azure by using Microsoft Entra ID including passwordless Quickstart: Azure Blob Storage client library for Python Assign share-level permissions Quickstart: Azure Blob Storage client library for .NET Deploy Azure Virtual Desktop
az role definition	Provide access to Key Vault keys, certificates, and secrets with an Azure role-based access control Transfer an Azure subscription to a different Microsoft Entra directory List Azure role definitions Use Azure role-based access control for Kubernetes Authorization Remediate non-compliant resources with Azure Policy Assign Azure roles using Azure CLI Manage access to Azure Machine Learning workspaces Use Azure Key Vault to pass secure parameter value during deployment

Reference subgroup	Azure CLI article showing reference use
	Use Azure Key Vault to pass secure parameter value during Bicep deployment
	Disable basic authentication in App Service deployments

az search

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az search admin-key	Connect to Azure AI Search using key authentication
	2 - Create and load Search Index with .NET
	Manage your Azure AI Search service with the Azure CLI
	2 - Create and load Search Index with JavaScript
az search private-endpoint-connection	Manage your Azure AI Search service with the Azure CLI
az search query-key	Connect to Azure AI Search using key authentication
	2 - Create and load Search Index with .NET
	Manage your Azure AI Search service with the Azure CLI
	2 - Create and load Search Index with JavaScript
az search service	Quickstart: Deploy Azure AI Search service using Terraform
	2 - Create and load Search Index with .NET
	Manage your Azure AI Search service with the Azure CLI
	2 - Create and load Search Index with JavaScript
az search shared-private-link-resource	Make outbound connections through a shared private link
	Manage your Azure AI Search service with the Azure CLI

az security

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az security atp cosmosdb	Enable Microsoft Defender for Azure Cosmos DB
az security atp storage	Enable Microsoft Defender for Storage (classic)
az security pricing	Enable Microsoft Defender for Storage (classic)
	Connect Azure Arc-enabled servers to Microsoft Defender for Cloud
az security workspace-setting	Connect Azure Arc-enabled servers to Microsoft Defender for Cloud

az servicebus

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az servicebus migration	Migrate existing Azure Service Bus standard namespaces to the premium tier
az servicebus namespace	Azure Service Bus client library for Python - version 7.11.4
	Tutorial - Use PaaS services with an Azure Kubernetes Service (AKS) cluster
	Enable partitioning for an Azure Service Bus Premium namespace
	Use the Azure CLI to create a Service Bus namespace and a queue
	Grant managed identity the access to Event Grid destination
	Use Azure CLI to create a Service Bus topic and subscriptions to the topic
	Configure the minimum TLS version for a Service Bus namespace
	Migrate an application to use passwordless connections with Azure Service Bus
	Quickstart: Deploy an event-driven application to Azure Spring Apps

Reference subgroup	Azure CLI article showing reference use
	Export IoT data to Service Bus
az servicebus namespace authorization-rule keys	Use the Azure CLI to create a Service Bus namespace and a queue
	Use Azure CLI to create a Service Bus topic and subscriptions to the topic
	Quickstart: Deploy an event-driven application to Azure Spring Apps
az servicebus queue	Tutorial - Use PaaS services with an Azure Kubernetes Service (AKS) cluster
	Enable dead lettering on message expiration for Azure Service Bus queues and subscriptions
	Get message counters
	Enable message sessions for an Azure Service Bus queue or a subscription
	Enable duplicate message detection for an Azure Service Bus queue or a topic
	Event delivery with a managed identity
	Enable auto forwarding for Azure Service Bus queues and subscriptions
	Use the Azure CLI to create a Service Bus namespace and a queue
	Enable partitioning in Azure Service Bus basic or standard
	Quickstart: Deploy an event-driven application to Azure Spring Apps
az servicebus queue authorization-rule	Tutorial - Use PaaS services with an Azure Kubernetes Service (AKS) cluster
	Create and delete routes and endpoints by using the Azure CLI
	Export IoT data to Service Bus
az servicebus queue authorization-rule keys	Tutorial - Use PaaS services with an Azure Kubernetes Service (AKS) cluster
	Create and delete routes and endpoints by using the Azure

Reference subgroup	Azure CLI article showing reference use
	CLI
	Export IoT data to Service Bus
az servicebus topic	Enable duplicate message detection for an Azure Service Bus queue or a topic
	Enable partitioning in Azure Service Bus basic or standard
	Grant managed identity the access to Event Grid destination
	Use Azure CLI to create a Service Bus topic and subscriptions to the topic
az servicebus topic authorization-rule keys	Create and delete routes and endpoints by using the Azure CLI
az servicebus topic subscription	Enable dead lettering on message expiration for Azure Service Bus queues and subscriptions
	Get message counters
	Enable message sessions for an Azure Service Bus queue or a subscription
	Enable auto forwarding for Azure Service Bus queues and subscriptions
	Use Azure CLI to create a Service Bus topic and subscriptions to the topic
az servicebus topic subscription rule	Message replication and cross-region federation
	Use Azure CLI to create a Service Bus topic and subscriptions to the topic

az sf

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az sf cluster	Quickstart: Deploy Linux containers to Service Fabric
	Create a Service Fabric cluster using Azure Resource Manager

Reference subgroup	Azure CLI article showing reference use
	Deploy a Linux Service Fabric cluster into an Azure virtual network
	Tutorial: Package and deploy containers as a Service Fabric application using Yeoman
	Tutorial: Deploy a Java application to a Service Fabric cluster in Azure
	Create a secure Service Fabric Linux cluster via the Azure CLI
	Enable disk encryption for Azure Service Fabric cluster nodes in Windows
	Enable disk encryption for Azure Service Fabric cluster nodes in Linux

az sig

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az sig	Store and share images in an Azure Compute Gallery
	Create a VM from a generalized image version
	Create a gallery for storing and sharing resources
	Share a gallery with all users in a subscription or tenants (preview)
	Tutorial: Create and use a custom image for Virtual Machine Scale Sets with the Azure CLI
	Share gallery resources across subscriptions and tenants with RBAC
	Tutorial: Create a custom image of an Azure VM with the Azure CLI
	How to use Azure Image Builder triggers to set up an automatic image build
	Create a VM using a specialized image version
	Create a Linux image and distribute it to an Azure Compute Gallery by using the Azure CLI
az sig gallery-application	Create and deploy VM Applications
az sig gallery-application version	Create and deploy VM Applications

Reference subgroup	Azure CLI article showing reference use
az sig image-definition	Hibernating virtual machines Create an image definition and an image version
	Deploy a VM with trusted launch enabled
	Create a VM from a generalized image version
	Trusted Launch FAQ
	Tutorial: Create and use a custom image for Virtual Machine Scale Sets with the Azure CLI
	Tutorial: Create a custom image of an Azure VM with the Azure CLI
	How to use Azure Image Builder triggers to set up an automatic image build
	Create a VM using a specialized image version
	Create a Linux image and distribute it to an Azure Compute Gallery by using the Azure CLI
az sig image-version	Store and share images in an Azure Compute Gallery Create an image definition and an image version
	Deploy a VM with trusted launch enabled
	Create a VM from a generalized image version
	Create an Azure Image Builder Bicep or ARM template JSON template
	Tutorial: Create and use a custom image for Virtual Machine Scale Sets with the Azure CLI
	Disable or remove the Linux Agent from VMs and images
	Tutorial: Create a custom image of an Azure VM with the Azure CLI
	Create a Linux image and distribute it to an Azure Compute Gallery by using the Azure CLI
	List, update, and delete gallery resources
az sig share	Share a gallery with all users in a subscription or tenants (preview) Share images using a community gallery
	List, update, and delete gallery resources

az signalr

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az signalr	Use private endpoints for Azure SignalR Service
	How to scale an Azure SignalR Service instance?
	Quickstart: Use an ARM template to deploy Azure SignalR Service
	How to send events from Azure SignalR Service to Event Grid
	Integrate Azure Digital Twins with Azure SignalR Service
	Create a SignalR Service with an App Service
	Create a SignalR Service
	Create a web app that uses SignalR Service and GitHub authentication
az signalr key	Azure SignalR Service authentication
	Connection strings in Azure SignalR Service
	Create a SignalR Service with an App Service
	Create a SignalR Service
	Create a web app that uses SignalR Service and GitHub authentication

az snapshot

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az snapshot	Create a snapshot of a virtual hard disk
	Change the disk type of an Azure managed disk
	Create an incremental snapshot for managed disks
	Migrate from in-tree storage class to CSI drivers on Azure Kubernetes Service (AKS)
	Copy an incremental snapshot to a new region

Reference subgroup	Azure CLI article showing reference use
	Create a virtual machine from a snapshot with CLI
	Tutorial - Manage Azure disks with the Azure CLI
	Export/Copy a snapshot to a storage account in different region with CLI
	Create a managed disk from a snapshot with CLI (Linux)
	Copy snapshot of a managed disk to same or different subscription with CLI

az spatial-anchors-account

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az spatial-anchors-account	Run the sample app: Android - Android Studio (Java or C++/NDK)
	Tutorial: Step-by-step instructions to create a new Android app using Azure Spatial Anchors
	Run the sample app: iOS - Xcode (Swift or Objective-C)
az spatial-anchors-account key	Run the sample app: Android - Android Studio (Java or C++/NDK)
	Tutorial: Step-by-step instructions to create a new Android app using Azure Spatial Anchors
	Run the sample app: iOS - Xcode (Swift or Objective-C)

az spring

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az spring	Quickstart: Deploy your first application to Azure Spring Apps
	Reliability in Azure Spring Apps

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Use a managed identity to invoke Azure Functions from an Azure Spring Apps app
	Deploy Azure Spring Apps in a virtual network
	Start or stop your Azure Spring Apps service instance
	Connect Azure Spring Apps to Key Vault using managed identities
	Quickstart: Deploy your first web application to Azure Spring Apps
	Quickstart: Deploy RESTful API application to Azure Spring Apps
	Use Application Insights Java In-Process Agent in Azure Spring Apps
	Quickstart: Deploy microservice applications to Azure Spring Apps
az spring api-portal	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Use API portal for VMware Tanzu
	Quickstart: Configure single sign-on for applications using the Azure Spring Apps Enterprise plan
az spring apm	How to configure APM integration and CA certificates
az spring app	Azure Spring Apps API breaking changes
	Troubleshoot common Azure Spring Apps issues
	Quickstart: Deploy your first application to Azure Spring Apps
	Tutorial: Use a managed identity to invoke Azure Functions from an Azure Spring Apps app
	Tutorial: Create a passwordless connection to a database service via Service Connector
	Enable system-assigned managed identity for an application in Azure Spring Apps
	Connect Azure Spring Apps to Key Vault using managed identities

Reference subgroup	Azure CLI article showing reference use
	Set up a staging environment in Azure Spring Apps
	Quickstart: Deploy your first web application to Azure Spring Apps
	Quickstart: Deploy RESTful API application to Azure Spring Apps
az spring app custom-domain	Map an existing custom domain to Azure Spring Apps
	Expose applications to the internet with TLS Termination at Application Gateway
	Enable ingress-to-app TLS for an application
	Expose applications with end-to-end TLS in a virtual network
	Map DNS names to applications in multiple Azure Spring Apps service instances in the same virtual network
	Quickstart: Map a custom domain to Azure Spring Apps with the Standard consumption and dedicated plan
az spring app deployment	Set up a staging environment in Azure Spring Apps
	Deploy an application with a custom container image
	How to Deploy Spring Boot applications from Azure CLI
	Connect an Azure Cosmos DB database to your application in Azure Spring Apps
	Capture heap dump and thread dump manually and use Java Flight Recorder in Azure Spring Apps
	Bind an Azure Database for PostgreSQL to your application in Azure Spring Apps
	Connect Azure Cache for Redis to your application in Azure Spring Apps
	Connect an Azure Database for MySQL instance to your application in Azure Spring Apps
az spring app identity	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate a Python application to use passwordless connections with Azure SQL Database

Reference subgroup	Azure CLI article showing reference use
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Migrate an application to use passwordless connections with Azure Cosmos DB for NoSQL
	Enable system-assigned managed identity for an application in Azure Spring Apps
	Migrate a .NET application to use passwordless connections with Azure SQL Database
	Migrate a Java application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Service Bus
	Migrate an application to use passwordless connections with Azure Event Hubs
	Migrate an application to use passwordless connections with Azure Event Hubs for Kafka
az spring app-insights	Use Application Insights Java In-Process Agent in Azure Spring Apps
az spring application-accelerator	Use VMware Tanzu Application Accelerator with the Azure Spring Apps Enterprise plan
az spring application-accelerator customized-accelerator	Use VMware Tanzu Application Accelerator with the Azure Spring Apps Enterprise plan
az spring application-accelerator predefined-accelerator	Use VMware Tanzu Application Accelerator with the Azure Spring Apps Enterprise plan
az spring application-configuration-service	Quickstart: Deploy microservice applications to Azure Spring Apps
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Use Application Configuration Service for Tanzu
	Migrate an Azure Spring Apps Basic or Standard plan instance to the Enterprise plan
	Quickstart: Configure single sign-on for applications using the Azure Spring Apps Enterprise plan

Reference subgroup	Azure CLI article showing reference use
az spring application-configuration-service git repo	Quickstart: Deploy microservice applications to Azure Spring Apps
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Use Application Configuration Service for Tanzu
	Migrate an Azure Spring Apps Basic or Standard plan instance to the Enterprise plan
az spring application-live-view	Use Application Live View with the Azure Spring Apps Enterprise plan
az spring build-service	How to deploy polyglot apps in the Azure Spring Apps Enterprise plan
az spring build-service build	How to deploy polyglot apps in the Azure Spring Apps Enterprise plan
	Deploy web static files
az spring build-service builder	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Quickstart: Deploy your first Java Native Image application to Azure Spring Apps
az spring build-service builder buildpack-binding	Use Application Insights Java In-Process Agent in Azure Spring Apps
	How to configure APM integration and CA certificates
	Migrate an Azure Spring Apps Basic or Standard plan instance to the Enterprise plan
	Quickstart: Monitor applications end-to-end
az spring certificate	Map an existing custom domain to Azure Spring Apps
	Expose applications with end-to-end TLS in a virtual network
az spring component	Use Application Configuration Service for Tanzu
	Troubleshoot VMware Spring Cloud Gateway
az spring component instance	Stream Azure Spring Apps managed component logs in real time
	Stream Azure Spring Apps managed component logs in real time

Reference subgroup	Azure CLI article showing reference use
	time
az spring config-server	Quickstart: Enable and disable Spring Cloud Config Server in Azure Spring Apps
az spring config-server git	Use Azure Spring Apps CI/CD with GitHub Actions
	Quickstart: Set up Spring Cloud Config Server for Azure Spring Apps
	Tutorial: Deploy apps to Azure Spring Apps using Jenkins and the Azure CLI
	Quickstart: Enable and disable Spring Cloud Config Server in Azure Spring Apps
az spring connection	Service Connector internals
	Use a managed identity to connect Azure SQL Database to an app deployed to Azure Spring Apps
	Connect an Azure Cosmos DB database to your application in Azure Spring Apps
	Connect Azure Cache for Redis to your application in Azure Spring Apps
	Quickstart: Load application secrets using Key Vault
	Get connection configurations added by Service Connector
	Quickstart: Integrate Azure Spring Apps with Azure Database for MySQL
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
	Quickstart: Create a service connection in Azure Spring Apps with the Azure CLI
az spring connection create	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Use a managed identity to connect Azure SQL Database to an app deployed to Azure Spring Apps
	Tutorial: Deploy a Spring application to Azure Spring Apps with a passwordless connection to an Azure database
	Connect an Azure Cosmos DB database to your application in Azure Spring Apps

Reference subgroup	Azure CLI article showing reference use
	Migrate a Java application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Service Bus
	Migrate an application to use passwordless connections with Azure Event Hubs for Kafka
	Migrate an application to use passwordless connections with Azure Database for MySQL
	Bind an Azure Database for PostgreSQL to your application in Azure Spring Apps
	Connect Azure Cache for Redis to your application in Azure Spring Apps
az spring container-registry	How to deploy polyglot apps in the Azure Spring Apps Enterprise plan
az spring dev-tool	Quickstart: Deploy microservice applications to Azure Spring Apps
	Use Application Live View with the Azure Spring Apps Enterprise plan
	Use VMware Tanzu Application Accelerator with the Azure Spring Apps Enterprise plan
	Configure Tanzu Dev Tools in the Azure Spring Apps Enterprise plan
az spring eureka-server	Tutorial: Authenticate client with Spring Cloud Gateway on Azure Spring Apps
	Quickstart: Enable and disable Eureka Server in Azure Spring Apps
az spring gateway	Quickstart: Deploy microservice applications to Azure Spring Apps
	Use Spring Cloud Gateway
	Configure VMware Spring Cloud Gateway
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Quickstart: Load application secrets using Key Vault

Reference subgroup	Azure CLI article showing reference use
	Use API portal for VMware Tanzu
	Quickstart: Integrate with Azure Database for PostgreSQL and Azure Cache for Redis
	Quickstart: Configure single sign-on for applications using the Azure Spring Apps Enterprise plan
	Quickstart: Monitor applications end-to-end
	Quickstart: Set request rate limits
az spring gateway route-config	Quickstart: Deploy microservice applications to Azure Spring Apps
	Use Spring Cloud Gateway
	Configure VMware Spring Cloud Gateway
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Quickstart: Integrate Azure OpenAI
	Use API portal for VMware Tanzu
	Quickstart: Configure single sign-on for applications using the Azure Spring Apps Enterprise plan
	Quickstart: Set request rate limits
az spring service-registry	Quickstart: Deploy microservice applications to Azure Spring Apps
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan
	Use Tanzu Service Registry
	Migrate an Azure Spring Apps Basic or Standard plan instance to the Enterprise plan
	Quickstart: Configure single sign-on for applications using the Azure Spring Apps Enterprise plan
az spring storage	How to enable your own persistent storage in Azure Spring Apps

az spring-cloud

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az spring-cloud app	Tutorial: Deploy a Spring Boot app connected to Apache Kafka on Confluent Cloud with Service Connector in Azure Spring Apps
az spring-cloud connection create	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate a .NET application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Event Hubs
	Migrate an application to use passwordless connections with Azure Queue Storage
	Tutorial: Deploy a Spring Boot app connected to Apache Kafka on Confluent Cloud with Service Connector in Azure Spring Apps

az sql

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az sql	Quickstart: Create a single database - Azure SQL Database
az sql db	Quickstart: Create a single database - Azure SQL Database
	Serverless compute tier for Azure SQL Database
	High availability for Azure SQL Database
	Copy a transactionally consistent copy of a database in Azure SQL Database
	Quickstart: Import a bacpac file to a database in Azure SQL Database or Azure SQL Managed Instance
	Connect to and query Azure SQL Database using .NET and Entity Framework Core
	Connect to and query Azure SQL Database using .NET and the Microsoft.Data.SqlClient library
	Change automated backup settings for Azure SQL Database

Reference subgroup	Azure CLI article showing reference use
	Use Java and JDBC with Azure SQL Database
	Hyperscale secondary replicas
az sql db geo-backup	Configure geo replication and backup restore for transparent data encryption with database level customer-managed keys
az sql db ledger-digest-uploads	Enable automatic digest storage
	Quickstart: Create a database in Azure SQL Database with ledger enabled
az sql db ltr-backup	Manage Azure SQL Database long-term backup retention
	Manage Azure SQL Managed Instance long-term backup retention
az sql db ltr-policy	Manage Azure SQL Database long-term backup retention
az sql db op	Scale single database resources in Azure SQL Database
	How to manage a Hyperscale database
	Monitor and scale a single database in Azure SQL Database using the Azure CLI
az sql db replica	Tutorial: Configure active geo-replication and failover (Azure SQL Database)
	Hyperscale secondary replicas
	Configure a license-free standby replica (preview) for Azure SQL Database
	Configure isolated access to a Hyperscale named replica
	Create a zone-redundant Hyperscale database
	Configure geo replication and backup restore for transparent data encryption with database level customer-managed keys
	Configure active geo-replication for a single database in Azure SQL Database using the Azure CLI
	Configure active geo-replication for a pooled database in Azure SQL Database using the Azure CLI
az sql db str-policy	Change automated backup settings for Azure SQL Database
az sql db tde	PowerShell and Azure CLI: Enable Transparent Data Encryption with customer-managed key from Azure Key Vault

Reference subgroup	Azure CLI article showing reference use
az sql db tde key	Identity and key management for TDE with database level customer-managed keys
az sql db-arc	Restore to a point-in-time
az sql elastic-pool	Configure maintenance window in Azure SQL Database
	Enable Always Encrypted with secure enclaves in Azure SQL Database
	Migrate Azure SQL Database to availability zone support
	Move a database in SQL Database in a SQL elastic pool using the Azure CLI
	Working with Hyperscale elastic pools using command-line tools
	Scale an elastic pool in Azure SQL Database using the Azure CLI
	Configure active geo-replication for a pooled database in Azure SQL Database using the Azure CLI
	Add an Azure SQL Database elastic pool to a failover group using the Azure CLI
az sql elastic-pool op	Scale elastic pool resources in Azure SQL Database
az sql failover-group	Tutorial: Implement a geo-distributed database (Azure SQL Database)
	Configure a failover group for a group of databases in Azure SQL Database using the Azure CLI
	Add a database to a failover group using the Azure CLI
	Add an Azure SQL Database elastic pool to a failover group using the Azure CLI
az sql instance-failover-group-arc	Configure failover group - CLI
az sql instance-pool	Create an instance pool (preview) - Azure SQL Managed Instance
az sql mi	Managed identities in Microsoft Entra for Azure SQL
	Configure public endpoints in Azure SQL Managed Instance
	Stop and start an instance - Azure SQL Managed Instance
	vCore purchasing model - Azure SQL Managed Instance
	Move Azure SQL Managed Instance across subnets

Reference subgroup	Azure CLI article showing reference use
	Create server with Microsoft Entra-only authentication enabled in Azure SQL
	Create an instance pool (preview) - Azure SQL Managed Instance
	Change automated backup settings for Azure SQL Managed Instance
	User-initiated manual failover on SQL Managed Instance
	Configure minimal TLS version in Azure SQL Managed Instance
az sql mi ad-only-auth	Microsoft Entra-only authentication with Azure SQL
	Tutorial: Enable Microsoft Entra-only authentication with Azure SQL
az sql mi dtc	Distributed Transaction Coordinator (DTC) for Azure SQL Managed Instance
az sql mi key	Rotate the Transparent data encryption (TDE) protector
	Azure CLI script to enable transparent data encryption using your own key
az sql mi op	Move Azure SQL Managed Instance across subnets
	Monitoring Azure SQL Managed Instance management operations
	Canceling Azure SQL Managed Instance management operations
az sql mi server-configuration-option	CREATE EXTERNAL TABLE AS SELECT (CETAS) (Transact-SQL)
az sql mi start-stop-schedule	Stop and start an instance - Azure SQL Managed Instance
az sql mi tde-key	Rotate the Transparent data encryption (TDE) protector
	Azure CLI script to enable transparent data encryption using your own key
az sql mi-arc	Storage Configuration
	Deploy a SQL Managed Instance enabled by Azure Arc
	High Availability with SQL Managed Instance enabled by Azure Arc
	Perform a point-in-time Restore
	Enable transparent data encryption on SQL Managed Instance enabled by Azure Arc (preview)

Reference subgroup	Azure CLI article showing reference use
	Configure SQL Managed Instance enabled by Azure Arc
	Connect to SQL Managed Instance enabled by Azure Arc
	Create SQL Managed Instance enabled by Azure Arc using Azure Data Studio
	Resource sync
	Quickstart: Deploy Azure Arc-enabled data services - indirectly connected mode - Azure CLI
az sql mi-arc endpoint	View logs and metrics using Kibana and Grafana
az sql midb	Restore a database in Azure SQL Managed Instance to a previous point in time
	Manage Azure SQL Managed Instance long-term backup retention
	Create an instance pool (preview) - Azure SQL Managed Instance
	Configure a ledger database
	Create an Azure SQL Managed Instance using the Azure CLI
az sql midb copy	Copy or move a database - Azure SQL Managed Instance
az sql midb ledger-digest-uploads	Enable automatic digest storage
az sql midb log-replay	Migrate databases from SQL Server by using Log Replay Service - Azure SQL Managed Instance
az sql midb ltr-backup	Manage Azure SQL Managed Instance long-term backup retention
az sql midb ltr-policy	Manage Azure SQL Managed Instance long-term backup retention
az sql midb move	Copy or move a database - Azure SQL Managed Instance
az sql midb short-term-retention-policy	Change automated backup settings for Azure SQL Managed Instance
az sql midb-arc	Perform a point-in-time Restore
az sql server	Quickstart: Create a single database - Azure SQL Database
	Azure SQL connectivity settings
	Managed identities in Microsoft Entra for Azure SQL
	sp_invoke_external_rest_endpoint (Transact-SQL)

Reference subgroup	Azure CLI article showing reference use
	Use Java and JDBC with Azure SQL Database
	PowerShell and Azure CLI: Enable Transparent Data Encryption with customer-managed key from Azure Key Vault
	Create server with Microsoft Entra-only authentication enabled in Azure SQL
	Outbound firewall rules for Azure SQL Database and Azure Synapse Analytics
	Create a single database and configure a firewall rule using the Azure CLI
	Create an Azure SQL Database server with a user-assigned managed identity
az sql server ad-admin	Transfer an Azure subscription to a different Microsoft Entra directory
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Tutorial: Connect a function app to Azure SQL with managed identity and SQL bindings
	Tutorial: Connect an App Service app to SQL Database on behalf of the signed-in user
	Migrate a Java application to use passwordless connections with Azure SQL Database
az sql server ad-only-auth	Microsoft Entra-only authentication with Azure SQL
	Tutorial: Enable Microsoft Entra-only authentication with Azure SQL
	Tutorial: Connect an App Service app to SQL Database on behalf of the signed-in user
az sql server audit-policy	Auditing using managed identity
az sql server dns-alias	PowerShell for DNS Alias to Azure SQL Database
az sql server firewall-rule	Quickstart: Create a single database - Azure SQL Database
	Azure SQL Database and Azure Synapse IP firewall rules
	Use Java and JDBC with Azure SQL Database
	Create a single database and configure a firewall rule using the Azure

Reference subgroup	Azure CLI article showing reference use
	CLI
	Quickstart: Create a Hyperscale database in Azure SQL Database
	Use Spring Data R2DBC with Azure SQL Database
	Backup an Azure SQL single database to an Azure storage container using the Azure CLI
	Import a BACPAC file into a database in SQL Database using the Azure CLI
	Quickstart: Create a Synapse SQL pool with Azure CLI
	Quickstart: Create a database in Azure SQL Database with ledger enabled
az sql server key	PowerShell and Azure CLI: Enable Transparent Data Encryption with customer-managed key from Azure Key Vault
	Rotate the Transparent data encryption (TDE) protector
	Remove a Transparent Data Encryption (TDE) protector using PowerShell
az sql server outbound-firewall-rule	Outbound firewall rules for Azure SQL Database and Azure Synapse Analytics
az sql server tde-key	PowerShell and Azure CLI: Enable Transparent Data Encryption with customer-managed key from Azure Key Vault
	Rotate the Transparent data encryption (TDE) protector
	Remove a Transparent Data Encryption (TDE) protector using PowerShell
az sql server-arc backups-policy	Manage automated backups - SQL Server enabled by Azure Arc
az sql server-arc extension feature-flag	Operate SQL Server enabled by Azure Arc with least privilege (preview)
az sql virtual-cluster	Resolve private domain names in Azure SQL Managed Instance
az sql vm	Automate management with the Windows SQL Server IaaS Agent extension
	Register Windows SQL Server VM with SQL IaaS Agent extension
	Enable Microsoft Entra authentication for SQL Server on Azure VMs

Reference subgroup	Azure CLI article showing reference use
	Change the license model for a SQL virtual machine in Azure
	SQL best practices assessment for SQL Server on Azure VMs
	In-place change of SQL Server edition - SQL Server on Azure VMs
	Use the Azure portal to configure a multiple-subnet availability group for SQL Server on Azure VMs
	Use PowerShell or Az CLI to configure an availability group for SQL Server on Azure VM
	SQL Server IaaS Agent extension for Linux
	Register Linux SQL Server VM with SQL IaaS Agent extension
az sql vm group	Use the Azure portal to configure a multiple-subnet availability group for SQL Server on Azure VMs
	Use PowerShell or Az CLI to configure an availability group for SQL Server on Azure VM
az sql vm group ag-listener	Use PowerShell or Az CLI to configure an availability group for SQL Server on Azure VM

az ssh

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az ssh	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	How to connect using Secure Shell (SSH) and sign on to an Azure virtual machine running Windows
	Create a Linux virtual machine on Azure
	Connect to Azure Operator Nexus Kubernetes cluster
	Tutorial: Install a LEMP stack on an Azure Linux VM

az sshkey

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az sshkey	Quick steps: Create and use an SSH public-private key pair for Linux VMs in Azure
	How to use SSH keys with Windows on Azure
	Quickstart: Deploy an Azure Kubernetes Service (AKS) cluster using an ARM template
	Quickstart: Deploy an Azure Kubernetes Service (AKS) cluster using Bicep
	Generate and store SSH keys with the Azure CLI
	Quickstart: Deploy an Azure Kubernetes Service (AKS) cluster using the Bicep extensibility Kubernetes provider (preview)
	Install the Kubernetes Event-driven Autoscaling (KEDA) add-on using an ARM template
	Quickstart: deploy a Kubernetes cluster using an Azure Resource Manager template

az stack

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az stack group	Deployment stacks (Preview)
	Quickstart: Create and deploy a deployment stack with Bicep
	Tutorial: use deployment stack with Bicep (Preview)
	Quickstart: Create and deploy a deployment stack with Bicep from template specs (Preview)
az stack mg	Deployment stacks (Preview)
az stack sub	Deployment stacks (Preview)

az stack-hci

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az stack-hci arc-setting	Azure Arc extension management on Azure Stack HCI
az stack-hci cluster	Azure Hybrid Benefit for Azure Stack HCI
	License Windows Server VMs on Azure Stack HCI
	Azure verification for VMs
	Azure Benefits on Azure Stack HCI (22H2 and earlier)
az stack-hci extension	Azure Arc extension management on Azure Stack HCI

az stack-hci-vm

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az stack-hci-vm	Create Arc virtual machines on Azure Stack HCI
	Manage Arc VMs on Azure Stack HCI
	Deploy Trusted launch for Azure Arc VMs on Azure Stack HCI, version 23H2
az stack-hci-vm disk	Manage resources for Arc VM on your Azure Stack HCI
az stack-hci-vm image	Create Azure Stack HCI VM image using Azure Marketplace images
	Create Azure Stack HCI VM image using images in a local share
	Create Azure Stack HCI VM image using image in Azure Storage account
az stack-hci-vm network Inet	Create logical networks for Azure Stack HCI
	Create Kubernetes clusters using Azure CLI
az stack-hci-vm network nic	Create network interfaces for Arc virtual machines on Azure Stack HCI
az stack-hci-vm storagepath	Create storage path for Azure Stack HCI

Reference subgroup	Azure CLI article showing reference use
	Use Container Storage Interface (CSI) disk drivers in AKS enabled by Azure Arc

az staticwebapp

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az staticwebapp	Quickstart: Building your first static site using the Azure CLI
	Tutorial: Create a serverless chat app with Azure Web PubSub service and Azure Static Web Apps
	Automate tasks with Azure CLI
az staticwebapp appsettings	Configure application settings for Azure Static Web Apps
az staticwebapp apptings	Configure application settings for Azure Static Web Apps
	Tutorial: Create a serverless chat app with Azure Web PubSub service and Azure Static Web Apps
az staticwebapp enterprise-edge	Tutorial: Configure Azure Front Door for Azure Static Web Apps
	Enterprise-grade edge
az staticwebapp secrets	Deploy a static web app with Azure Static Web Apps CLI

az storage

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az storage account	Learn to use Bash with the Azure CLI
	How to use variables in Azure CLI commands
	How to manage Azure resource groups with the Azure CLI
	Learn Azure CLI syntax differences in Bash, PowerShell and Cmd

Reference subgroup	Azure CLI article showing reference use
	Populate variables for use in scripts
	Create a resource using a service principal
	Delete Azure resources at scale using a script
	Considerations for running the Azure CLI in a PowerShell environment
	Configure Azure Storage firewalls and virtual networks
	Create a storage account
az storage account blob-inventory-policy	Enable Azure Storage blob inventory reports
az storage account blob-service-properties	Configure a lifecycle management policy
	Change feed support in Azure Blob Storage
	Configure object replication for block blobs
	Enable and manage blob versioning
	Enable soft delete for blobs
	Enable and manage soft delete for containers
	Perform a point-in-time restore on block blob data
	Manage block blobs with Azure CLI
az storage account encryption-scope	Create and manage encryption scopes
az storage account file-service-properties	SMB file shares in Azure Files
	Enable soft delete on Azure file shares
az storage account hns-migration	Upgrade Azure Blob Storage with Azure Data Lake Storage Gen2 capabilities
az storage account keys	Learn to use Bash with the Azure CLI
	Create a resource using a service principal
	Manage storage account access keys

Reference subgroup	Azure CLI article showing reference use
	Create and use a volume with Azure Files in Azure Kubernetes Service (AKS)
	Store Terraform state in Azure Storage
	Quickstart: Import a bacpac file to a database in Azure SQL Database or Azure SQL Managed Instance
	Mount SMB Azure file share on Linux
	Mount an Azure file share in Azure Container Instances
	Troubleshoot Azure Files identity-based authentication and authorization issues (SMB)
	Step 1: Upload image data in the cloud with Azure Storage
<code>az storage account local-user</code>	Connect to Azure Blob Storage by using the SSH File Transfer Protocol (SFTP)
<code>az storage account management-policy</code>	Configure a lifecycle management policy
	Migrate from diagnostic settings storage retention to Azure Storage lifecycle management
<code>az storage account migration</code>	Change the redundancy configuration for a storage account
<code>az storage account network-rule</code>	Configure Azure Storage firewalls and virtual networks
	Configure a firewall for serverless compute access
	Configuring Azure Files network endpoints
	Configuring Azure File Sync network endpoints
	Restrict network access to PaaS resources with virtual network service endpoints using the Azure CLI
	Manage data exfiltration to Azure Storage accounts with virtual network service endpoint policies using the Azure CLI
<code>az storage account or-policy</code>	Configure object replication for block blobs
<code>az storage account or-policy rule</code>	Configure object replication for block blobs
<code>az storage blob</code>	Learn to use Bash with the Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Deploy and score a machine learning model by using an online endpoint
	Troubleshooting online endpoints deployment and scoring
	Host a static website in Azure Storage
	Quickstart: Create, download, and list blobs with Azure CLI
	Download a Windows VHD from Azure
	Set a blob's access tier
	Configure immutability policies for blob versions
	Archive a blob
	Use blob index tags to manage and find data on Azure Blob Storage
az storage blob copy	Set a blob's access tier
	Archive a blob
	Rehydrate an archived blob to an online tier
	Export/Copy a managed disk to a storage account using the Azure CLI
	Export/Copy a snapshot to a storage account in different region with CLI
	Manage block blobs with Azure CLI
	Create an image and use a user-assigned managed identity to access files in an Azure storage account
	Convert append blobs and page blobs into block blobs
az storage blob directory	Enable and manage blob versioning
az storage blob immutability-policy	Configure immutability policies for blob versions
az storage blob metadata	Manage block blobs with Azure CLI
az storage blob service-properties	Host a static website in Azure Storage
	Azure Front Door: Deploy custom domain

Reference subgroup	Azure CLI article showing reference use
	Deploy OSDU Admin UI on top of Azure Data Manager for Energy
az storage blob tag	Use blob index tags to manage and find data on Azure Blob Storage
	Manage block blobs with Azure CLI
az storage container	Azure CLI configuration
	Learn to use Bash with the Azure CLI
	Configure anonymous read access for containers and blobs
	Prevent Shared Key authorization for an Azure Storage account
	Store Terraform state in Azure Storage
	Quickstart: Create, download, and list blobs with Azure CLI
	Tutorial: Upload and analyze a file with Azure Functions and Blob Storage
	Using linked and nested templates when deploying Azure resources
	Configure object replication for block blobs
	Step 1: Upload image data in the cloud with Azure Storage
az storage container immutability-policy	Configure immutability policies for blob versions
	Configure immutability policies for containers
	Quickstart: Create a database in Azure SQL Database with ledger enabled
az storage container legal-hold	Configure immutability policies for containers
az storage container metadata	Learn to use Bash with the Azure CLI
	Manage blob containers using Azure CLI
az storage container policy	Use Azure Blob storage Shared Access Signatures to restrict access to data in HDInsight
az storage container-rm	Configure immutability policies for blob versions

Reference subgroup	Azure CLI article showing reference use
az storage cors	Quickstart - Get started with 3D Scenes Studio (preview) for Azure Digital Twins
	Build 3D scenes with 3D Scenes Studio (preview) for Azure Digital Twins
az storage directory	Quickstart: Create and use an Azure file share Access Azure file shares using Microsoft Entra ID with Azure Files OAuth over REST
	Transactional replication with Azure SQL Managed Instance
	Mount a file share to a Python function app using Azure CLI
az storage entity	Measure Azure Cosmos DB for NoSQL performance with a benchmarking framework
az storage file	Quickstart: Create and use an Azure file share Access Azure file shares using Microsoft Entra ID with Azure Files OAuth over REST
	Transactional replication with Azure SQL Managed Instance
	Restore Azure file shares with the Azure CLI
az storage fs	Manage and restore soft-deleted blobs Manage directories and files in Azure Data Lake Storage Gen2 via the Azure CLI
az storage fs access	Use Azure CLI to manage ACLs in Azure Data Lake Storage Gen2 403 Access Denied authorization error when the sticky bit is enabled in ADLS Gen2
az storage fs directory	Manage directories and files in Azure Data Lake Storage Gen2 via the Azure CLI
az storage fs file	Manage directories and files in Azure Data Lake Storage Gen2 via the Azure CLI
az storage fs service-properties	Enable soft delete for blobs
az storage message	Connect Azure Functions to Azure Storage using command line tools
	Tutorial: Deploy an event-driven job with Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Tutorial: Deploy a background processing application with Azure Container Apps
	Connect your Java function to Azure Storage
az storage queue	Connect Azure Functions to Azure Storage using command line tools
	Tutorial: Deploy an event-driven job with Azure Container Apps
	Tutorial: Deploy a background processing application with Azure Container Apps
	Quickstart: Route custom events to Azure Queue storage via Event Grid using Azure CLI
	Connect your Java function to Azure Storage
	Choose how to authorize access to queue data with Azure CLI
az storage share	Create an SMB Azure file share
	Create and use a volume with Azure Files in Azure Kubernetes Service (AKS)
	Mount an Azure file share in Azure Container Instances
	Mount NFS Azure file share on Linux
	Access Azure file shares using Microsoft Entra ID with Azure Files OAuth over REST
	Enable automatic HTTPS with Caddy in a sidecar container
	Move an Azure Synapse Analytics workspace from one region to another
	Mount a file share to a Python function app using Azure CLI
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Files
	Back up Azure file shares with Azure CLI
az storage share-rm	Create an SMB Azure file share
	Quickstart: Create and use an Azure file share
	Tutorial: Create an Azure Files volume mount in Azure Container Apps

Reference subgroup	Azure CLI article showing reference use
	Enable soft delete on Azure file shares
	How to enable your own persistent storage in Azure Spring Apps with the Standard consumption and dedicated plan
az storage table	How to use the Azure SDK for Go with Azure Table

az storage-mover

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az storage-mover	Create an Azure Storage Mover resource

az storageSync

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az storageSync sync-group	Deploy Azure File Sync
az storageSync sync-group cloud-endpoint	Deploy Azure File Sync
az storageSync sync-group server-endpoint	Deploy Azure File Sync

az stream-analytics

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az stream-analytics input	Quickstart: Create an Azure Stream Analytics job using the Azure CLI
az stream-analytics job	Quickstart: Create an Azure Stream Analytics job by using an ARM template
	Quickstart: Create an Azure Stream Analytics job using Terraform

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create an Azure Stream Analytics job using the Azure CLI
az stream-analytics output	Quickstart: Create an Azure Stream Analytics job using the Azure CLI
az stream-analytics transformation	Quickstart: Create an Azure Stream Analytics job using the Azure CLI

az support

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az support services	Create an Azure support ticket in Azure CLI
az support services problem-classifications	Create an Azure support ticket in Azure CLI
az support tickets	Create an Azure support ticket in Azure CLI
az support tickets communications	Create an Azure support ticket in Azure CLI

az synapse

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az synapse role assignment	Move an Azure Synapse Analytics workspace from one region to another
az synapse spark pool	Move an Azure Synapse Analytics workspace from one region to another
az synapse sql pool	Quickstart: Create a Synapse SQL pool with Azure CLI
az synapse workspace	Move an Azure Synapse Analytics workspace from one region to another
	Quickstart: Create an Azure synapse workspace with Azure CLI
az synapse workspace firewall-rule	Azure SQL Database and Azure Synapse IP firewall rules

Reference subgroup	Azure CLI article showing reference use
	Move an Azure Synapse Analytics workspace from one region to another
	Quickstart: Create an Azure synapse workspace with Azure CLI
az synapse workspace managed-identity	Move an Azure Synapse Analytics workspace from one region to another

az tag

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az tag	Learn Azure CLI syntax differences in Bash, PowerShell and Cmd
	Apply tags with Azure CLI
	Enable Microsoft Defender for Storage (classic)
	Apply inventory tagging to Azure Arc-enabled servers
	Tutorial: Route policy state change events to Event Grid with Azure CLI
	How to peer payment HSM virtual networks

az term

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az term	Troubleshoot common Azure Spring Apps issues
	Quickstart: Deploy your first application to Azure Spring Apps
	Quickstart: Deploy RESTful API application to Azure Spring Apps
	Quickstart: Deploy microservice applications to Azure Spring Apps
	Quickstart: Build and deploy apps to Azure Spring Apps using the Enterprise plan

Reference subgroup	Azure CLI article showing reference use
	Enterprise plan in Azure Marketplace
	Use Tanzu Build Service
	Migrate an Azure Spring Apps Basic or Standard plan instance to the Enterprise plan
	Collect Spring Cloud Resilience4J Circuit Breaker Metrics with Micrometer (Preview)
	Use Application Live View with the Azure Spring Apps Enterprise plan

az ts

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az ts	How to use Azure Resource Manager (ARM) deployment templates with Azure CLI
	Azure Resource Manager template specs in Bicep
	Azure Resource Manager template specs
	Quickstart: Create and deploy a template spec with Bicep
	Quickstart: Create and deploy template spec
	Tutorial: Create a template spec with linked templates
	Tutorial: Create Azure portal forms for a template spec
	Quickstart: Create and deploy a deployment stack with Bicep from template specs (Preview)
	Tutorial: Deploy a template spec as a linked template

az tsi

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az tsi access-policy	Authentication and authorization for Azure Time Series Insights API
az tsi environment	Create an Azure Time Series Insights Gen2 environment using the Azure CLI
az tsi environment gen2	Create an Azure Time Series Insights Gen2 environment using the Azure CLI

az upgrade

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Install Azure CLI on Windows
	Install Azure CLI on macOS
	Create an Azure service principal with Azure CLI
	How to update the Azure CLI
	How to use variables in Azure CLI commands
	Overview and Prerequisites
	Create an Azure support ticket in Azure CLI
	Use the Azure REST API with Azure CLI
	Supported Kubernetes versions in Azure Kubernetes Service (AKS)
	Quickstart: Create a single database - Azure SQL Database

az version

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
Azure CLI	Install Azure CLI on Windows
	Install Azure CLI on macOS
	Create an Azure service principal with Azure CLI

Reference subgroup	Azure CLI article showing reference use
	Tips for using the Azure CLI successfully
	How to use variables in Azure CLI commands
	Prepare your environment for the Azure CLI
	Overview and Prerequisites
	Create an Azure support ticket in Azure CLI
	Considerations for running the Azure CLI in a PowerShell environment
	Use the Azure REST API with Azure CLI

az vm

[Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az vm	Tips for using the Azure CLI successfully
	Output formats for Azure CLI commands
	How to use variables in Azure CLI commands
	Get VM information with queries
	Create a virtual machine on a virtual network
	Learn Azure CLI syntax differences in Bash, PowerShell and Cmd
	Log in to a Windows virtual machine in Azure by using Microsoft Entra ID including passwordless
	Availability sets overview
	Create a private Azure Kubernetes Service (AKS) cluster
	Change the size of a virtual machine
az vm aem	New Version of Azure VM extension for SAP solutions
	Standard Version of Azure VM extension for SAP solutions
az vm application	Create and deploy VM Applications

Reference subgroup	Azure CLI article showing reference use
az vm availability-set	Implement Oracle Data Guard on a Linux-based Azure virtual machine
	Create a complete Linux virtual machine with the Azure CLI
	Deploy VMs to proximity placement groups using Azure CLI
	Tutorial: Manually install IBM WebSphere Application Server Network Deployment traditional on Azure virtual machines
	Migrate a Linux virtual machine from unmanaged disks to managed disks
	Tutorial: Manually install Oracle WebLogic Server on Azure Virtual Machines
	Deploy an IPv6 dual stack application in Azure virtual network using Azure CLI
	Tutorial: Configure availability groups for SQL Server on RHEL virtual machines in Azure
	Load balancing on multiple IP configurations using Azure CLI
	Create a public load balancer with IPv6 using Azure CLI
az vm boot-diagnostics	Repair a Windows VM by using the Azure Virtual Machine repair commands
	Repair a Linux VM by using the Azure Virtual Machine repair commands
	Troubleshoot Resource Manager deployment issues with creating a new Linux virtual machine in Azure
	Troubleshoot a Linux VM by attaching the OS disk to a recovery VM with the Azure CLI
az vm disk	Using Azure ultra disks
	Azure Disk Encryption scenarios on Linux VMs
	Create an Oracle Database in an Azure VM
	Add a disk to a Linux VM
	Preview - Increase IOPS and throughput limits for Azure Premium SSDs and Standard SSD/HDDs
	How to detach a data disk from a Linux virtual machine
	Enable on-demand bursting

Reference subgroup	Azure CLI article showing reference use
	Tutorial - Manage Azure disks with the Azure CLI
	Configure LVM and RAID on encrypted devices
	Tutorial: Modify a Virtual Machine Scale Set using Azure CLI
az vm encryption	Handling special cases when moving virtual machines to resource group or subscription
	Azure Disk Encryption scenarios on Windows VMs
	Create and configure a key vault for Azure Disk Encryption on a Windows VM
	Azure Disk Encryption scenarios on Linux VMs
	Unlocking an encrypted disk for offline repair
	Quickstart: Create and encrypt a Windows VM with the Azure CLI
	Creating and configuring a key vault for Azure Disk Encryption
	Configure LVM and RAID on encrypted devices
	Unlocking an encrypted Linux disk for offline repair
	Azure Disk Encryption with Microsoft Entra ID for Windows VMs (previous release)
az vm extension	Log in to a Windows virtual machine in Azure by using Microsoft Entra ID including passwordless
	Manage Azure Monitor Agent
	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	How to connect using Secure Shell (SSH) and sign on to an Azure virtual machine running Windows
	Hibernating virtual machines
	Troubleshoot SSH connections to an Azure Linux VM that fails, errors out, or is refused
	Create a Linux virtual machine on Azure
	Use the Azure Custom Script Extension Version 2 with Linux virtual machines
	Azure Disk Encryption scenarios on Windows VMs

Reference subgroup	Azure CLI article showing reference use
	Migrate the existing agent-based hybrid workers to extension-based hybrid workers
az vm extension image	Virtual machine extensions and features for Linux
	Troubleshooting guidance for the Azure Monitor agent on Linux virtual machines and scale sets
	Troubleshooting guidance for the Azure Monitor agent on Windows virtual machines and scale sets
	Enable Azure VM extensions using the Azure CLI
	New Version of Azure VM extension for SAP solutions
	Update Azure Network Watcher extension to the latest version
	Develop ARM templates for cloud consistency
az vm host	Deploy VMs and scale sets to dedicated hosts
	Add Azure Dedicated Host to an Azure Kubernetes Service (AKS) cluster
az vm host group	Deploy VMs and scale sets to dedicated hosts
	Add Azure Dedicated Host to an Azure Kubernetes Service (AKS) cluster
az vm identity	Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)
	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	Use an Azure managed identity to authenticate to an Azure container registry
	Use the Linux diagnostic extension 4.0 to monitor metrics and logs
	Connect with Managed Identity to Azure Database for PostgreSQL
	Connect with managed identity to Azure Database for PostgreSQL - Flexible Server
	Configure managed identities for Azure resources on an Azure VM using Azure CLI
	Tutorial: Use Azure Key Vault with a virtual machine in .NET

Reference subgroup	Azure CLI article showing reference use
	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for .NET
az vm image	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for Python
az vm image	Support for generation 2 VMs on Azure
	Azure VM Image Builder overview
az vm image	Handling special cases when moving virtual machines to resource group or subscription
	Azure Virtual Machine Scale Set automatic OS image upgrades
	Find Azure Marketplace image information using the Azure CLI
	Create an Azure Image Builder Bicep or ARM template JSON template
	Tutorial: Create and Manage Linux VMs with the Azure CLI
	Overview of Red Hat Enterprise Linux images
	Trusted Launch FAQ
	Oracle VM images and their deployment on Microsoft Azure
az vm image terms	Handling special cases when moving virtual machines to resource group or subscription
	Find Azure Marketplace image information using the Azure CLI
	Red Hat Enterprise Linux bring-your-own-subscription Gold Images in Azure
	Use dashboards to visualize Azure Databricks metrics
	Deploy an Azure Kubernetes application by using an ARM template
	Move a Marketplace Azure Virtual Machine to another subscription
	Deploy an Azure Kubernetes application programmatically by using Azure CLI
	Deploy and manage applications from Azure Marketplace on Azure Arc-enabled Kubernetes clusters
az vm nic	Azure Instance Metadata Service
	Associate a public IP address to a virtual machine

Reference subgroup	Azure CLI article showing reference use
	Dissociate a public IP address from an Azure VM
	High availability for SAP HANA on Azure VMs on SUSE Linux Enterprise Server
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server for SAP applications
	Back up and recover Oracle Database on an Azure Linux VM by using Azure Backup
	How to create a Linux virtual machine in Azure with multiple network interface cards
	High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux
	Tutorial: Connect to Virtual Machine Scale Set instances using the Azure CLI
	High availability for SAP NetWeaver on Azure VMs on SUSE Linux Enterprise Server with NFS on Azure Files
az vm repair	Repair a Windows VM by using the Azure Virtual Machine repair commands
	Azure Linux virtual machine fails to boot after applying kernel changes
	How to reset a local Linux password on Azure VMs
	Troubleshoot Linux VM boot issues due to fstab errors
	Repair a Linux VM by using the Azure Virtual Machine repair commands
	Use Azure Linux Auto Repair (ALAR) to fix a Linux VM
	Troubleshoot Resource Manager deployment issues with creating a new Linux virtual machine in Azure
	Troubleshoot SSH connection issues in Azure Linux VM due to permission and ownership issues
az vm run-command	Run scripts in your Windows VM by using action Run Commands
	How to connect using Secure Shell (SSH) and sign on to an Azure virtual machine running Windows
	Quick steps: Create and use an SSH public-private key pair for Linux VMs in Azure
	How to use SSH keys with Windows on Azure

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create a Windows virtual machine with the Azure CLI
	Run scripts in your Linux VM by using action Run Commands
	Troubleshoot issues with the Log Analytics agent for Windows
	Certificate rotation in Azure Kubernetes Service (AKS)
	Run scripts in your Windows VM by using managed Run Commands
	Run scripts in your Linux VM by using managed Run Commands
az vm secret	Tutorial: Use TLS/SSL certificates to secure a web server
	Tutorial - How to use cloud-init to customize a Linux virtual machine in Azure on first boot
az vm unmanaged-disk	Attach an unmanaged disk to a VM for offline repair
az vm user	Troubleshoot SSH connections to an Azure Linux VM that fails, errors out, or is refused
	VMAccess Extension for Linux
	How to reset a local Linux password on Azure VMs
	Use Azure Policy to restrict extensions installation on Linux VMs

az vmss

[\[+\] Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az vmss	Change the size of a virtual machine
	Azure Virtual Machine Scale Set agents
	Troubleshoot Azure Kubernetes Service cluster/node in a failed state
	Explore Azure Hybrid Benefit for Windows VMs
	Azure Virtual Machine Scale Set automatic OS image upgrades
	Delete a VM and attached resources

Reference subgroup	Azure CLI article showing reference use
	Troubleshoot issues with the Log Analytics agent for Windows
	Overview of autoscale with Azure Virtual Machine Scale Sets
	Modify a Virtual Machine Scale Set
	Create a Virtual Machine Scale Set that uses Availability Zones
az vmss application	Create and deploy VM Applications
az vmss disk	Tutorial: Create and use disks with Virtual Machine Scale Set with the Azure CLI
az vmss encryption	Encrypt OS and attached data disks in a Virtual Machine Scale Set with the Azure CLI
	Enable disk encryption for Azure Service Fabric cluster nodes in Windows
	Enable disk encryption for Azure Service Fabric cluster nodes in Linux
az vmss extension	Manage Azure Monitor Agent
	Azure Virtual Machine Scale Set agents
	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	Troubleshoot issues with the Log Analytics agent for Windows
	VMAccess Extension for Linux
	Use the Linux diagnostic extension 4.0 to monitor metrics and logs
	Automatic Extension Upgrade for VMs and Scale Sets in Azure
	Azure Key Vault virtual machine extension for Windows
	Using Application Health extension with Virtual Machine Scale Sets
	Tutorial: Create an application gateway with URL path-based redirection using the Azure CLI
az vmss identity	Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)
	Log in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH

Reference subgroup	Azure CLI article showing reference use
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Use the Linux diagnostic extension 4.0 to monitor metrics and logs
	Send Prometheus data to Azure Monitor by using managed identity authentication
	Configure managed identities for Azure resources on a virtual machine scale set using Azure CLI
	Create a chaos experiment that uses an agent-based fault with the Azure CLI
az vmss rolling-upgrade	Azure Virtual Machine Scale Set automatic OS image upgrades
az vmss run-command	Troubleshoot issues with the Log Analytics agent for Windows
	Certificate rotation in Azure Kubernetes Service (AKS)
	Manage SSH for secure access to Azure Kubernetes Service (AKS) nodes
	Tunnel connectivity issues
	Troubleshoot Node Not Ready failures if there are expired certificates
	Troubleshoot node not ready failures caused by CSE errors

az vmware

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az vmware datastore	Attach Azure NetApp Files datastores to Azure VMware Solution hosts
az vmware datastore netapp-volume	Attach Azure NetApp Files datastores to Azure VMware Solution hosts
az vmware private-cloud	Configure customer-managed key encryption at rest in Azure VMware Solution

az webapp

Reference subgroup	Azure CLI article showing reference use
az webapp	Quickstart: Deploy an ASP.NET web app
	Quickstart: Deploy a Python (Django or Flask) web app to Azure App Service
	Use Key Vault references as app settings in Azure App Service and Azure Functions
	Deploy a Node.js web app in Azure
	Deploy files to App Service
	Tutorial: Host a RESTful API with CORS in Azure App Service
	Inbound and outbound IP addresses in Azure App Service
	Configure a Node.js app for Azure App Service
	Configure a Linux Python app for Azure App Service
	Migrate custom software to Azure App Service using a custom container
az webapp auth	Tutorial: Authenticate and authorize users end-to-end in Azure App Service
	Work with OAuth tokens in Azure App Service authentication
	Customize sign-in and sign-out in Azure App Service authentication
	Tutorial: Connect an App Service app to SQL Database on behalf of the signed-in user
	Manage the API and runtime versions of App Service authentication
az webapp config	Configure an App Service app
	Use Key Vault references as app settings in Azure App Service and Azure Functions
	Configure a custom container for Azure App Service
	Configure a Node.js app for Azure App Service
	Configure a Linux Python app for Azure App Service
	Migrate custom software to Azure App Service using a custom container

Reference subgroup	Azure CLI article showing reference use
	Deploy your app to Azure App Service using FTP/S
	Connect to and query Azure SQL Database using Python and the pyodbc driver
	Open an SSH session to a container in Azure App Service
	Use CI/CD to deploy a Python web app to Azure App Service on Linux
az webapp config access-restriction	Set up Azure App Service access restrictions
	Application Gateway integration
	Tutorial: Create a highly available multi-region app in Azure App Service
az webapp config appsettings	Configure an App Service app
	Configure a Node.js app for Azure App Service
	OS and runtime patching in Azure App Service
	Tutorial: Connect to a PostgreSQL Database from Java Tomcat App Service without secrets using a managed identity
	Tutorial: Using Service Connector to build a Django app with Postgres on Azure App Service
	Tutorial: Connect to a MySQL Database from Java JBoss EAP App Service with passwordless connection
az webapp config apptings	Configure an App Service app
	Quickstart: Deploy a Python (Django or Flask) web app to Azure App Service
	Deploy files to App Service
	Tutorial: Host a RESTful API with CORS in Azure App Service
	Configure a custom container for Azure App Service
	Configure a Node.js app for Azure App Service
	Use a TLS/SSL certificate in your code in Azure App Service
	Migrate custom software to Azure App Service using a custom container

Reference subgroup	Azure CLI article showing reference use
	Connect to and query Azure SQL Database using Python and the pyodbc driver
	Tutorial: Use a managed identity to connect Key Vault to an Azure web app in .NET
az webapp config backup	Back up and restore your app in Azure App Service
az webapp config connection-string	Configure an App Service app
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Tutorial: Connect an App Service app to SQL Database on behalf of the signed-in user
	Tutorial: Connect a Node.js web app with Azure Cosmos DB for MongoDB vCore
az webapp config container	Deployment Best Practices
	Configure a custom container for Azure App Service
	Migrate custom software to Azure App Service using a custom container
	Tutorial: Create a multi-container (preview) app in Web App for Containers
	Continuous deployment with custom containers in Azure App Service
	Create an ASP.NET Core app in a Docker container in App Service from Azure Container Registry
az webapp config hostname	How-to: build a real-time collaborative whiteboard using Azure Web PubSub and deploy it to Azure App Service
az webapp config snapshot	Back up and restore your app in Azure App Service
az webapp config storage-account	Bring dependencies or third party library to Azure Functions
	Mount a file share to a Python function app using Azure CLI
az webapp connection	Service Connector internals

Reference subgroup	Azure CLI article showing reference use
	Quickstart: Create a service connection in App Service with the Azure CLI
	Get connection configurations added by Service Connector
az webapp connection create	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Connect to and query Azure SQL Database using .NET and Entity Framework Core
	Connect to and query Azure SQL Database using .NET and the Microsoft.Data.SqlClient library
	Connect to and query Azure SQL Database using Node.js and mssql npm package
	Tutorial: Deploy a web application connected to Azure Blob Storage with Service Connector
	Migrate an application to use passwordless connections with Azure Blob Storage
	Create and deploy a Flask Python web app to Azure with system-assigned managed identity
	Tutorial: Create a passwordless connection to a database service via Service Connector
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Tutorial: Connect a web app to Azure App Configuration with Service Connector
az webapp cors	Tutorial: Host a RESTful API with CORS in Azure App Service
az webapp deployment	Quickstart: Deploy a Python (Django or Flask) web app to Azure App Service
	Deploy your app to Azure App Service using FTP/S
	Configure deployment credentials for Azure App Service
az webapp deployment container	Migrate custom software to Azure App Service using a custom container
	Continuous deployment with custom containers in Azure App Service
az webapp deployment	Use CI/CD with GitHub Actions to deploy a Python web app to Azure

Reference subgroup	Azure CLI article showing reference use
github-actions	App Service on Linux
az webapp deployment slot	Set up staging environments in Azure App Service
	Tutorial: Create a highly available multi-region app in Azure App Service
az webapp deployment source	Quickstart: Deploy a Python (Django or Flask) web app to Azure App Service
	Provision and publish a bot
	Local Git deployment to Azure App Service
	Run your app in Azure App Service directly from a ZIP package
	Step 1: Upload image data in the cloud with Azure Storage
	Disable basic authentication in App Service deployments
	Azure SignalR Service authentication
	Tutorial: Visualize real-time sensor data from your Azure IoT hub in a web application
	Use CI/CD with GitHub Actions to deploy a Python web app to Azure App Service on Linux
	How to use Azure SignalR Service with Azure Application Gateway
az webapp deployment user	Tutorial: Host a RESTful API with CORS in Azure App Service
	Tutorial: Use a managed identity to connect Key Vault to an Azure web app in .NET
	Create a PHP web app in Azure App Service
	Configure deployment credentials for Azure App Service
	Azure SignalR Service authentication
	Tutorial: Troubleshoot an App Service app with Azure Monitor
	Tutorial: Visualize real-time sensor data from your Azure IoT hub in a web application
	Tutorial: Deploy a PHP and Azure Database for MySQL - Flexible Server app on Azure App Service
	Create a web app that uses SignalR Service and GitHub authentication

Reference subgroup	Azure CLI article showing reference use
az webapp identity	How to use managed identities for App Service and Azure Functions Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Configure a custom container for Azure App Service
	Migrate custom software to Azure App Service using a custom container
	Connect to and query Azure SQL Database using Python and the pyodbc driver
	Tutorial: Use a managed identity to connect Key Vault to an Azure web app in .NET
	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Use system-assigned managed identities to access Azure Cosmos DB data
	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for .NET
	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for Python
az webapp log	Quickstart: Deploy a Python (Django or Flask) web app to Azure App Service Enable diagnostics logging for apps in Azure App Service
	Deploy a Node.js web app in Azure
	Configure a custom container for Azure App Service
	Configure a Node.js app for Azure App Service
	Configure a Linux Python app for Azure App Service
	Migrate custom software to Azure App Service using a custom container
	Deploy a containerized Flask or FastAPI web app on Azure App Service
	Azure Functions Java developer guide
	Configure a PHP app for Azure App Service
az webapp log deployment	Disable basic authentication in App Service deployments

Reference subgroup	Azure CLI article showing reference use
az webapp traffic-routing	Set up staging environments in Azure App Service
az webapp vnet-integration	Enable virtual network integration in Azure App Service
	Tutorial: Deploy Django app with App Service and Azure Database for PostgreSQL - Flexible Server
	Tutorial: Create a secure n-tier app in Azure App Service
	Tutorial: Isolate back-end communication in Azure App Service with Virtual Network integration
	Tutorial: Connect an App Services Web app to Azure Database for MySQL - Flexible Server in a virtual network
	Tutorial: Create an Azure Database for PostgreSQL - Flexible Server instance with App Services Web App in virtual network

az webpubsub

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az webpubsub	Tutorial: Publish and subscribe messages using WebSocket API and Azure Web PubSub service SDK
	Tutorial: Create a chat app with Azure Web PubSub service
	Tutorial: Publish and subscribe messages between WebSocket clients using subprotocol
	Create a Web PubSub resource
	Build a real-time code-streaming app by using Socket.IO and host it on Azure
	How-to: build a real-time collaborative whiteboard using Azure Web PubSub and deploy it to Azure App Service
	Tutorial: Create a serverless chat app with Azure Web PubSub service and Azure Static Web Apps
	Tutorial: Visualize IoT device data from IoT Hub using Azure Web PubSub service and Azure Functions

Reference subgroup	Azure CLI article showing reference use
	How-to: Build a real-time collaborative whiteboard using Web PubSub for Socket.IO and deploy it to Azure App Service
	Quickstart: Publish messages using the Azure Web PubSub service SDK
az webpubsub client	Quickstart: Publish messages using the Azure Web PubSub service SDK
az webpubsub hub	Tutorial: Create a chat app with Azure Web PubSub service
	Event handler in Azure Web PubSub service
	Event notifications from clients
	How-to: build a real-time collaborative whiteboard using Azure Web PubSub and deploy it to Azure App Service
	Tutorial: Create a serverless chat app with Azure Web PubSub service and Azure Static Web Apps
az webpubsub key	Tutorial: Publish and subscribe messages using WebSocket API and Azure Web PubSub service SDK
	Tutorial: Create a chat app with Azure Web PubSub service
	Tutorial: Publish and subscribe messages between WebSocket clients using subprotocol
	Build a real-time code-streaming app by using Socket.IO and host it on Azure
	How-to: build a real-time collaborative whiteboard using Azure Web PubSub and deploy it to Azure App Service
	Tutorial: Create a serverless chat app with Azure Web PubSub service and Azure Static Web Apps
	Tutorial: Visualize IoT device data from IoT Hub using Azure Web PubSub service and Azure Functions
	How-to: Build a real-time collaborative whiteboard using Web PubSub for Socket.IO and deploy it to Azure App Service
	Quickstart: Publish messages using the Azure Web PubSub service SDK
az webpubsub replica	Geo-replication in Azure Web PubSub

az workloads

 Expand table

Reference subgroup	Azure CLI article showing reference use
az workloads	Tutorial: Use Azure CLI to create infrastructure for a distributed highly available (HA) SAP system with <i>Azure Center for SAP solutions</i> with customized resource names
	Quickstart: Use Azure CLI to create infrastructure for a distributed highly available (HA) SAP system with Azure Center for SAP solutions with customized resource names
az workloads sap-application-server-instance	Soft stop SAP systems, application server instances and HANA database
az workloads sap-database-instance	Soft stop SAP systems, application server instances and HANA database
az workloads sap-virtual-instance	Quickstart: Register an existing SAP system with Azure Center for SAP solutions with CLI
	Tutorial: Use Azure CLI to create infrastructure for a distributed highly available (HA) SAP system with <i>Azure Center for SAP solutions</i> with customized resource names
	Soft stop SAP systems, application server instances and HANA database
	Quickstart: Use Azure CLI to create infrastructure for a distributed highly available (HA) SAP system with Azure Center for SAP solutions with customized resource names
	Quickstart: Install software for a Distributed High-Availability (HA) SAP system and customized resource names with Azure Center for SAP solutions using Azure CLI
	Quickstart: Start and stop SAP systems from Azure Center for SAP solutions with CLI

Quoting differences between scripting languages

Article • 06/27/2024

When you work with Azure CLI commands, be aware of how your scripting language uses quotation marks and escapes characters. If you support scripts used in different shells, understanding quoting differences saves you valuable development hours.

To avoid unanticipated results with parameter values containing single or double quotes, or escape characters, here are a few suggestions:

White spaces and quotation marks

- If you provide a parameter value that contains white space, wrap the value in quotation marks.
- In Bash and PowerShell, if your variable value contains single quotes, wrap the value in double quotes, and vice-versa.
- In Bash, double quotes that are escaped, are treated as part of the string.
- In Windows Command Prompt, quotes inside variable values are treated as part of the value.

Here are a few examples:

Bash

Azure CLI

```
# Correct
myVariable="my string ' ' wrapped in double quotes"
myVariable='my string " " wrapped in single quotes'
myVariable="my string with escaped \" \" double quotes wrapped in double quotes"
# Wrong, escaped single quotes in Bash are not treated as part of the string
myVariable='my value with escaped \' \' single quotes wrapped in single quotes'
# after each example ...
echo $myVariable
```

Bash output for the correct examples is as follows:

Output

```
my string '' wrapped in double quotes
my string " " wrapped in single quotes
my string with escaped " " double quotes wrapped in double quotes
```

If you want the quotes included in the output, escape the variable like this: `echo \\$myVariable\``.

Output

```
echo \\$myVariable\
"my string '' wrapped in double quotes"
echo \\$myVariable\
'my string " " wrapped in single quotes'
echo \\$myVariable\
"my string with escaped " " double quotes wrapped in double quotes"
```

JSON strings

- If your command is only going to run on Bash (or Zsh), use single quotes to preserve the content inside the JSON string. Single quotes are necessary when supplying inline JSON values. For example, this JSON is correct in Bash: `'{"key": "value"}'`.
- If your command runs at a Windows Command Prompt, you must use double quotes. If the value contains double quotes, you must escape it. The equivalent of the above JSON string in Cmd.exe is `"{\"key\": \"value\"}"`.
- When working with JSON parameter values, consider using Azure CLI's `@<file>` convention and bypass the shell's interpretation mechanisms.

Azure CLI

```
az ad app create --display-name myName --native-app --required-
resource-accesses @manifest.json
```

If you are working in the PowerShell scripting language, see [Considerations for running the Azure CLI in a PowerShell scripting language - JSON strings](#).

Empty strings

- In PowerShell, if your value is an empty quotes string (''), use ''''.
- In Bash or PowerShell, if your value is an empty quotes string (), use ''''.

```
Azure CLI
```

```
# Correct in Bash
myVariable=''''

# Correct in PowerShell
$myVariable = ''
$myVariable = '''''
```

Space-separated values

Some Azure CLI commands take a list of space separated values. If the key name or value contains spaces, wrap the whole pair: "my key=my value". For example:

```
Azure CLI
```

```
az web app config app settings set --resource-group myResourceGroup --name
myWebAppName --settings "client id=id1" "my name=john"
```

When a CLI parameter states that it accepts a space-separated list, one of two formats is expected:

- Example of unquoted, space-separated list: --parameterName firstValue
secondValue
- Example of quoted space-separated list: --parameterName "firstValue"
"secondValue"

This example is a string with a space in it. It isn't a space-separated list: --

```
parameterName "firstValue secondValue"
```

Special characters

There are special characters in the PowerShell scripting language, such as at @. To run Azure CLI in PowerShell, add ` before the special character to escape it. You can also enclose the value in single or double quotes "/".

```
PowerShell
```

```
# The following three examples will work in PowerShell
--parameterName `@parameters.json
--parameterName '@parameters.json'
--parameterName "@parameters.json"

# This example will not work in PowerShell
--parameterName @parameters.json
```

Hyphen characters

If a parameter's value begins with a hyphen, Azure CLI tries to parse it as a parameter name. To parse it as value, use `=` to concatenate the parameter name and value: `--password="-VerySecret"`.

The `--query` parameter

When you use the `--query` parameter with a command, some characters of [JMESPath](#) need to be escaped in the shell.

Bash

These three commands are correct and equivalent in Bash:

Azure CLI

Here are two examples of *incorrect commands* in Bash:

```
Azure CLI

# Wrong, as the dash needs to be quoted in a JMESPath query
az version --query azure-cli
az version: error: argument --query: invalid jmespath_type value:
'azure-cli'

# Wrong, as the dash needs to be quoted in a JMESPath query, but quotes
# are interpreted by Bash
az version --query "azure-cli"
az version: error: argument --query: invalid jmespath_type value:
'azure-cli'
```

For more example comparisons between Bash, PowerShell, and Cmd, see [Query Azure CLI command output](#).

The `--debug` parameter

The best way to troubleshoot a quoting issue is to run the command with the `--debug` flag. This flag reveals the actual arguments received by the Azure CLI in [Python's syntax](#).

```
Azure CLI

# Correct
$ az '{"key":"value"}' --debug
>> Command arguments: ['{"key":"value"}', '--debug']

# Correct
$ az "{\"key\":\"value\"}" --debug
>> Command arguments: ['{"key":"value"}', '--debug']

# Wrong, as quotes and spaces are interpreted by Bash
$ az {"key": "value"} --debug
>> Command arguments: ['{key:', 'value}', '--debug']

# Wrong, as quotes are interpreted by Bash
$ az {"key":"value"} --debug
>> Command arguments: ['{key:value}', '--debug']
```

For more information on troubleshooting Azure CLI commands with `--debug`, see [Troubleshooting Azure CLI](#).

Scripting language rules

Here are quick links to scripting language rules as published by their respective organizations:

- Bash scripting language: [Bash quoting rules](#)
- PowerShell scripting language: [PowerShell Quoting Rules](#)
- Windows Command Prompt: [How-to: Escape Characters, Delimiters, and Quotes at the Windows command line](#)

Note

Due to a known issue in PowerShell, some extra escaping rules apply. For more information, see [Considerations for running the Azure CLI in a PowerShell](#)

See also

Find many more scripting language comparisons in these articles:

- [Learn syntax differences in Bash, PowerShell, and Cmd tutorial](#)
- [Query Azure CLI command output using a JMESPath query](#)
- [Considerations for running the Azure CLI in a PowerShell scripting language](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Tips for using the Azure CLI successfully

Article • 06/27/2024

Azure CLI is a command-line tool that allows you to configure and manage Azure resources from many shell environments. After you [choose your preferred shell environment](#) and [install the Azure CLI](#), use this article to discover useful tips on how to avoid common pitfalls and use the Azure CLI successfully.

To learn more about specific Azure CLI commands, see the [Azure CLI Reference list](#).

Choose a scripting language

The Azure CLI is an open source *tool* that you can run in many *scripting languages*.

- It's the scripting language that determines the quoting, escape character, and line continuation rules for Azure CLI commands.
- It can be challenging to copy Azure CLI command syntax *containing single or double quotes, escape characters, or line continuation characters* between languages.
- **Most Microsoft documentation for the Azure CLI is written and tested in Bash.**
- If PowerShell is your scripting language of choice, consider using the tool, [Azure PowerShell](#), which has native [PowerShell](#) scripting language functionality.

For more information on scripting language options for Azure CLI, see [Choose the right command-line tool](#).

FAQ

Here are some links to help you answer the most frequently asked Azure CLI questions:

[+] Expand table

Subject area with article link	Tip
Web Account Manager (WAM)	Information on WAM and how to disable it.
Configuration settings	List of available Azure CLI configuration settings, how to turn on/off Azure CLI features, and set default values.
Output formats	Instructions on how to store Azure resource IDs in variables, and protect

Subject area with article link	Tip
	secrets returned by Azure CLI commands.
Passing JSON values in a string	Quoting examples for JSON parameter values.
Support lifecycle	Information about supported operating systems, scripting languages and the end of service for Azure CLI versions and dependencies.
REST API	Alternative <code>az rest</code> command to use when a specific Azure CLI command isn't available for an Azure resource.
Azure CLI reference groups	A to Z list of Azure CLI reference groups with drill-down to subgroups and all Azure CLI commands.
Azure CLI articles	A to Z list of Quickstarts, How-to guides, and Tutorials that use Azure CLI reference commands. The list is grouped by reference group and subgroup for easy search.
Azure CLI samples	A to Z list of Samples, including those samples in the Azure-samples/azure-cli-samples GitHub repository.
Troubleshooting	Error cause and remediation tips.
What's new	Short summary of new Azure CLI features.

Pass values to another command

If the value is used more than once, assign it to a variable. Variables allow you to use values more than once or to create more general scripts. This example assigns an ID found by the `az vm list` command to a variable.

```
Azure CLI

# assign the list of running VMs to a variable
running_vm_ids=$(az vm list --resource-group MyResourceGroup --show-details \
    --query "[?powerState=='VM running'].id" --output tsv)

# verify the value of the variable
echo $running_vm_ids
```

If the value is used only once, consider piping. (Piping passes the output of one command as input to a second command.)

```
Azure CLI
```

```
az vm list --query "[?powerState=='VM running'].name" --output tsv | grep my_vm
```

For multi-value lists, consider the following options:

1. If you need more controls on the result, use a "for" loop:

Azure CLI

```
#!/usr/bin/env bash
for vmList in $(az vm list --resource-group MyResourceGroup --show-details \
--query "[?powerState=='VM running'].id" --output tsv); do
    echo stopping $vmList
    az vm stop --ids $vmList
    if [ $? -ne 0 ]; then
        echo "Failed to stop $vmList"
        exit 1
    fi
    echo $vmList stopped
done
```

1. Alternatively, use `xargs` and consider using the `-P` flag to run the operations in parallel for improved performance:

Azure CLI

```
az vm list --resource-group MyResourceGroup --show-details \
--query "[?powerState=='VM stopped'].id" \
--output tsv | xargs -I {} -P 10 az vm start --ids "{}"
```

1. Finally, Azure CLI has built-in support to process commands with multiple `--ids` in parallel to achieve the same effect of xargs. `@-` is used to get values from the pipe:

Azure CLI

```
az vm list --resource-group MyResourceGroup --show-details \
--query "[?powerState=='VM stopped'].id" \
--output tsv | az vm start --ids @-
```

For more information on using Bash constructs with the Azure CLI including loops, case statements, if..then..else, and error handling, see [Learn to use Bash with the Azure CLI](#).

Common update parameters

Azure CLI command groups often feature an update command. For instance, [Azure Virtual Machines](#) includes the `az vm update` command. Most update commands offer the three common parameters: `--add`, `--set`, and `--remove`.

The `--set` and `--add` parameters take a list of space-separated key-value pairs: `key1=value1 key2=value2`. To see what properties you can update, use a show command, such as [az vm show](#).

```
Azure CLI
```

```
az vm show --resource-group VMResources --name virtual-machine-01
```

To simplify the command, consider using a JSON string. For example, to attach a new data disk to a virtual machine, use the following value:

```
Azure CLI
```

```
az vm update --resource-group VMResources --name virtual-machine-01 \
--add storageProfile.dataDisks "{\"createOption\": \"Attach\",
\"managedDisk\": {
  \"id\": \"/subscriptions/00000000-0000-0000-0000-
0000000000/resourceGroups/yg/providers/Microsoft.Compute/disks/yg-disk\"},
  \"lun\": 1}"
```

Generic resource commands

A service you want to work with might not have Azure CLI support. You can use the [az resource](#) or [az rest](#) commands to work with these resources.

Concurrent execution

If you run Azure CLI commands concurrently on the same machine, write conflicts can happen if multiple Azure CLI commands write to the same Microsoft Authentication Library (MSAL) token cache.

To avoid potential failures, isolate the Azure CLI configuration folder for each script by setting environment variable `AZURE_CONFIG_DIR` for each script to a separate directory.

Azure CLI commands in that script save the configuration and token cache to the configured location instead of the default `~/.azure` folder.

Bash

Bash

```
export AZURE_CONFIG_DIR=/my/config/dir
```

Asynchronous operations

Operations in Azure can take a noticeable amount of time. For instance, configuring a virtual machine at a data center isn't instantaneous. Azure CLI waits until the command finishes to accept other commands. Many commands therefore offer a `--no-wait` parameter as shown here:

Azure CLI

```
az group delete --name MyResourceGroup --no-wait
```

When you delete a resource group, all the resources that belong to it are also removed. Removing these resources can take a long time. When you run the command with the `--no-wait` parameter, the console accepts new commands without interrupting the removal.

Many commands offer a wait option, pausing the console until some condition is met. The following example uses the `az vm wait` command to support creating independent resources in parallel:

Azure CLI

```
az vm create --resource-group VMResources --name virtual-machine-01 --image centos --no-wait
az vm create --resource-group VMResources --name virtual-machine-02 --image centos --no-wait

subscription=$(az account show --query "id" -o tsv)
vm1_id="/subscriptions/$subscription/resourceGroups/VMResources/providers/Microsoft.Compute/virtualMachines/virtual-machine-01"
vm2_id="/subscriptions/$subscription/resourceGroups/VMResources/providers/Microsoft.Compute/virtualMachines/virtual-machine-02"
az vm wait --created --ids $vm1_id $vm2_id
```

After both IDs are created, you can use the console again.

Script examples

Here are examples for using variables and looping through a list when working with Azure Virtual Machines. For in-depth examples on using Bash constructs with the Azure CLI including loops, case statements, if..then..else, and error handling, see [Learn to use Bash with the Azure CLI](#).

Use these scripts to save IDs to variables:

Bash

```
Azure CLI

ECHO OFF
SETLOCAL
FOR /F "tokens=* USEBACKQ" %%F IN (
    `az vm list --resource-group VMResources --show-details --query "[?powerState=='VM running'].id" --output tsv`)
) DO (
    SET "vm_ids=%F %vm_ids%" :: construct the id list
)
az vm stop --ids %vm_ids% :: CLI stops all VMs in parallel
```

Use these scripts to loop through a list:

Bash

```
Azure CLI

ECHO OFF
SETLOCAL
FOR /F "tokens=* USEBACKQ" %%F IN (
    `az vm list --resource-group VMResources --show-details --query "[?powerState=='VM running'].id" --output tsv`)
) DO (
    ECHO Stopping %%F
    az vm stop --ids %%F
)
```

See also

- [Troubleshooting the Azure CLI](#)

 Collaborate with us on
GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Choose the right Azure command-line tool

Article • 06/19/2023

When it comes to managing Azure, you have many options. This article compares the Azure CLI and Azure PowerShell language and gives a comparison of the shell environments on which they run.

Azure CLI, Azure PowerShell, and Azure Cloud Shell have overlapping functionality. Each operates differently, and the language is sometimes confused with the environment. Use this guide to determine which is the right tool for you.

What's the advantage of using an Azure command-line tool?

Azure runs on automation. Every action you take inside the portal translates somewhere to code being executed to read, create, modify, or delete resources.

Moving your workload to Azure lifts some of the administrative burden but not all. As your rate of adoption with Azure increases so will the overhead. Even though you don't have to worry about the data center, you still have to patch and troubleshoot Azure VMs, failover databases, and configure virtual networks.

By using the existing automation that runs Azure, command-line tools reduce that overhead.

What are Azure command-line tools?

Azure command-line tools automate routine operations, standardize database failovers, and pull data that provide powerful insight. Command-line tools not only give you the ability to scale your tasks in Azure, but they also make much easier to share. Sharing a script is much easier than a lengthy wiki page with time consuming screenshots.

Using an Azure command-line tool isn't always necessary, but it's a useful skill to have.

Azure CLI vs Azure PowerShell

Azure CLI and Azure PowerShell are command-line tools that enable you to create and manage Azure resources. Both are cross-platform, installable on Windows, macOS, and

Linux.

Azure CLI

- Cross-platform command-line interface, installable on Windows, macOS, Linux
- Runs in Windows PowerShell, Cmd, or Bash and other Unix shells.

Azure PowerShell

- Cross-platform PowerShell module, runs on Windows, macOS, Linux
- Requires Windows PowerShell or PowerShell

Different shell environments

Shell Environment	Azure CLI	Azure PowerShell
Cmd	Yes	
Bash	Yes	
Windows PowerShell	Yes	Yes
PowerShell	Yes	Yes

Windows PowerShell, PowerShell, Cmd, and Bash are shell environments. Your shell environment not only determines which tools you can use but also changes your command-line experience.

For example, for the line continuation character, Bash uses the backslash `\` while Windows PowerShell uses the backtick ```. The differences in the shell environment doesn't change how Azure CLI and Azure PowerShell operate. However, they do change your command-line experience.

Azure CLI has an installer that makes its commands executable in all four shell environments.

Azure PowerShell is set of cmdlets packaged as a PowerShell module named `Az`; not an executable. Windows PowerShell or PowerShell must be used to install the `Az` module.

Windows PowerShell is the standard scripting shell that comes preinstalled with most Windows operating systems. PowerShell is a stand-alone installation that uses .NET Core as its run time, allowing it to be installed on macOS, Linux, and Windows.

Key points:

- AzureRM is a PowerShell module that is still referenced for Azure administration with PowerShell. However, it has been replaced by Azure PowerShell and has an official retirement date of February 29 2024.

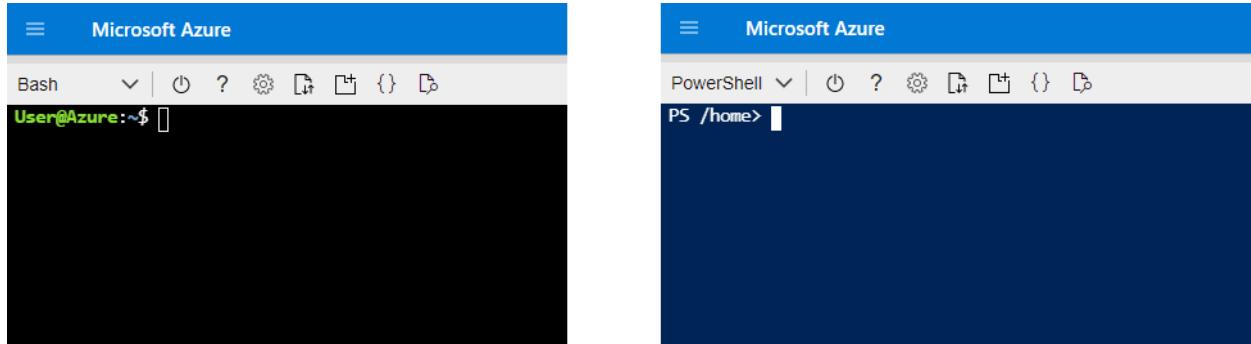
If you're using AzureRM, you can [migrate Azure PowerShell from AzureRM to Az.](#)

What about Azure Cloud Shell?

Azure Cloud Shell is a hosted shell environment that runs on an Ubuntu container.

Cloud Shell provides two shell environments: Bash (with Azure CLI preinstalled) and PowerShell (with Azure PowerShell preinstalled). A dropdown list at the top of the Cloud Shell window allows you to easily switch between the two environments.

Cloud Shell is accessible in a web browser and has integrations for [Windows Terminal](#) and [Visual Studio Code ↗](#).



ⓘ Note

Azure Cloud Shell may not always *immediately reflect* the most recent Azure PowerShell and Azure CLI releases as the publishing schedule for all three tools is different. However, Azure Cloud Shell is generally thought to always contain the most recent versions of both tools.

Which Azure command-line tool is right for you?

When picking the right tool, consider your past experience and current work environment.

Azure CLI syntax is similar to that of Bash scripting. If you work primarily with Linux systems, Azure CLI feels more natural.

Azure PowerShell is a PowerShell module. If you work primarily with Windows systems, Azure PowerShell is a natural fit. Commands follow a verb-noun naming scheme and data is returned as objects.

Choose the tool that uses your experience and shortens your learning curve. Take advantage of [Microsoft learning](#) to become proficient at managing Azure at the command line.

With that said, being open-minded will only improve your abilities. Use a different tool when it makes sense.

Key points:

- Feature parity for Azure services doesn't always exist between Azure CLI and Azure PowerShell.

Azure CLI vs Azure PowerShell: Side-by-side Command Comparison

Sign in, Subscription, and Location Commands:

Command	Azure CLI	Azure PowerShell
Sign in with Web Browser	az login	Connect-AzAccount
Get available subscriptions	az account list	Get-AzSubscription
Set Subscription	az account set --subscription <SubscriptionId>	Set-AzContext -Subscription <SubscriptionID>
List Azure Locations	az account list-locations	Get-AzLocation

Find Versions, Get Help, and View Command Help:

Command	Azure CLI	Azure PowerShell
Find Version	az --version	Get-InstalledModule -Name Az
Get Help	az --help	Get-Help
View Command Help	az vm --help	Get-Help -Name New-AzVM

Create a Resource Group, VM, and Storage Account:

Command	Azure CLI	Azure PowerShell
Create Resource Group	az group create --name <ResourceGroupName> --location eastus	New-AzResourceGroup -Name <ResourceGroupName> -Location eastus
Create Azure Virtual Machine	az vm create --resource-group myResourceGroup --name myVM --image UbuntuLTS --admin-username azureuser --admin-password '<Password>'	New-AzVM -ResourceGroupName <ResourceGroupName> -Name myVM -Image UbuntuLTS -Credential (Get-Credential)
Create Azure Storage Account	az storage account create --name <StorageAccountName> --resource-group <ResourceGroupName> --location eastus --sku Standard_LRS --kind StorageV2	New-AzStorageAccount -Name <StorageAccountName> -ResourceGroupName <ResourceGroupName> -Location eastus -SkuName Standard_LRS -Kind StorageV2

Manage Azure Virtual Machines:

Command	Azure CLI	Azure PowerShell
List VM	az vm list	Get-AzVM
Restart VM	az vm restart --name myVM --resource-group <ResourceGroupName>	Restart-AzVM -Name myVM -ResourceGroupName <ResourceGroupName>
Stop VM	az vm stop --name myVM --resource-group <ResourceGroupName>	Stop-AzVM -Name myVM -ResourceGroupName <ResourceGroupName>
Stop & Deallocate VM	az vm deallocate --name myVM --resource-group <ResourceGroupName>	Stop-AzVM -Name myVM -ResourceGroupName <ResourceGroupName>
Start VM	az vm start --name myVM --resource-group <ResourceGroupName>	Start-AzVM -Name myVM -ResourceGroupName <ResourceGroupName>
Delete VM	az vm delete --name myVM --resource-group <ResourceGroupName>	Remove-AzVM -Name myVM -ResourceGroupName <ResourceGroupName>

Select Properties and Change Output Formats:

Command	Azure CLI	Azure PowerShell
Show all subscription information	az account list --all	Get-AzSubscription Select-Object -Property *
Output as a Table	az account list -o table	Get-AzSubscription Format-Table
Output as JSON	az account show	Get-AzSubscription ConvertTo-Json

Key points:

- Azure CLI defaults to outputting a JSON string. Other format options can be found on the [Output formats for Azure CLI commands](#).
- Azure PowerShell defaults to outputting objects. To learn more about formatting in PowerShell, read the [Using Format Commands to Change Output View](#).

Next steps

Azure CLI:

- [Install the Azure CLI](#)

Azure PowerShell:

- [Install Azure PowerShell](#)

How-to use the Azure CLI in a Bash scripting language

Article • 06/27/2024

Azure CLI reference commands can execute in several [scripting languages](#). If you're new to Bash and also the Azure CLI, this article a great place to begin your learning journey. Work through this article much like you would a tutorial to learn how to use the Azure CLI in a Bash scripting language with ease.

In this article, you learn how to:

- ✓ Query results as JSON dictionaries or arrays
- ✓ Format output as JSON, table, or TSV
- ✓ Query, filter, and format single and multiple values
- ✓ Use if/exists/then and case syntax
- ✓ Use for loops
- ✓ Use grep, sed, paste, and bc commands
- ✓ Populate and use shell and environment variables

If you don't have an Azure subscription, create an [Azure free account](#) before you begin.

Starting Bash

Start Bash using [Azure Cloud Shell](#) or a [local install of the Azure CLI](#). This article assumes that you're running Bash either using Azure Cloud Shell or running Azure CLI locally in a docker container.

Querying dictionary results

A command that always returns only a single object returns a JSON dictionary. Dictionaries are unordered objects accessed with keys. For this article, we're going to start by querying the [Account](#) object using the [Account Show](#) command.

Azure CLI

```
az account show
az account show --output json # JSON is the default format
```

The following JSON dictionary output has some fields omitted for brevity, and identifying information is removed.

JSON

```
bash-5.1# az account show
{
  "environmentName": "AzureCloud",
  "isDefault": true,
  "managedByTenants": [],
  "name": "My test subscription",
  "state": "Enabled",
  "user": {
    "name": "user@contoso.com",
    "type": "user"
  }
}
```

Formatting the output as YAML

Use the `--output yaml` argument (or `-o yaml`) to format the output in [yaml](#) format, a plain-text data serialization format. YAML tends to be easier to read than JSON, and easily maps to that format. Some applications and CLI commands take YAML as configuration input, instead of JSON.

Azure CLI

```
az account show --output yaml
```

For more information about formatting the output as yaml, see [YAML output format](#).

Formatting the output as a table

Use the `--output table` argument (or `-o table`) to format the output as an ASCII table. Nested objects aren't included in table output, but can still be filtered as part of a query.

Azure CLI

```
az account show --output table
```

For more information about formatting the output as a table, see [Table output format](#).

Querying and formatting single values and nested values

The following queries demonstrate querying single values, including nested values in a JSON dictionary output. The final query in this set demonstrates formatting the output using the `-o tsv` argument. This argument returns the results as tab- and newline-separated values. This action is useful for removing quotation marks in the value returned - which is useful to consume the output into other commands and tools that need to process the text in some form (as is demonstrated later in this article).

Azure CLI

```
az account show --query name # Querying a single value
az account show --query name -o tsv # Removes quotation marks from the
output

az account show --query user.name # Querying a nested value
az account show --query user.name -o tsv # Removes quotation marks from the
output
```

Querying and formatting properties from arrays

The following query demonstrates getting properties in a JSON array. Get subscription properties, displayed as a table of subscriptions.

Azure CLI

```
az account list --query "[].{subscription_id:id, name:name,
isDefault:isDefault}" -o table
```

This query returns results similar to:

JSON

Subscription_id	Name
IsDefault	
-----	-----
11111111-3ddc-45ce-8334-c7b28a9e1c3a	C & L Azure developer experience
content projects	False
22222222-8f1c-409b-af1e-8e2e65d9b90a	DevCenter - Infrastructure - Dogfood
False	
33333333-c080-42a7-8973-1aa853ab4df3	Babel
False	

Querying and formatting multiple values, including nested values

To get more than one property, put expressions in square brackets [] (a multiselect list) as a comma-separated list. The following queries demonstrate querying multiple values in a JSON dictionary output, using multiple output formats.

Azure CLI

```
az account show --query [name,id,user.name] # return multiple values
az account show --query [name,id,user.name] -o table # return multiple
values as a table
```

For more information about returning multiple values, see [Get multiple values](#).

Renaming properties in a query

The following queries demonstrate the use of the {} (multiselect hash) operator to get a dictionary instead of an array when querying for multiple values. It also demonstrates renaming properties in the query result.

Azure CLI

```
az account show --query "{SubscriptionName: name, SubscriptionId: id,
UserName: user.name}" # Rename the values returned
az account show --query "{SubscriptionName: name, SubscriptionId: id,
UserName: user.name}" -o table # Rename the values returned in a table
```

For more information on renaming properties in a query, see [Rename properties in a query](#).

Querying boolean values

Boolean values are assumed to be true, so the "[?isDefault]" query syntax for the `az account list` command returns the current default subscription. To get the false values, you must use an escape character, such as \.

The following queries demonstrate querying all accounts in a subscription, potentially returning a JSON array if there are multiple subscriptions for a given account, and then querying for which account is the default subscription. It also demonstrates querying for the accounts that aren't the default subscription. These queries build on what you learned previously to filter and format the results. Finally, the final query demonstrates storing the query results in a variable.

Azure CLI

```

az account list
az account list --query "[?isDefault]" # Returns the default subscription
az account list --query "[?isDefault]" -o table # Returns the default
subscription as a table
az account list --query "[?isDefault].[name,id]" # Returns the name and id
of the default subscription
az account list --query "[?isDefault].[name,id]" -o table # Returns the name
and id of the default subscription as a table
az account list --query "[?isDefault].{SubscriptionName: name,
SubscriptionId: id}" -o table # Returns the name and id of the default
subscription as a table with friendly names

az account list --query "[?isDefault == `false`]" # Returns all non-
default subscriptions, if any
az account list --query "[?isDefault == `false`].name" -o table # Returns
all non-default subscriptions, if any, as a table

az account list --query "[?isDefault].id" -o tsv # Returns the subscription
id without quotation marks
subscriptionId=$(az account list --query "[?isDefault].id" -o tsv) # Captures the subscription id as a variable.
echo $subscriptionId # Returns the contents of the variable.
az account list --query "[? contains(name, 'Test')].id" -o tsv # Returns the
subscription id of a non-default subscription containing the substring
'Test'
subscriptionId=$(az account list --query "[? contains(name, 'Test')].id" -o
tsv) # Captures the subscription id as a variable.
az account set -s $subscriptionId # Sets the current active subscription

```

- For more information about filtering arrays and querying boolean values, see [Filter arrays with boolean expressions](#).
- For more information about using variables, see [How to use variables](#).
- For more information on working with subscriptions, see [Managing subscriptions](#).

Creating objects using variables and randomization

Setting a random value for use in subsequent commands

Setting and using a random value for use in variables allows you to run scripts multiple times without naming conflicts. Naming conflicts occur because a value must be unique across the service, or because an object you deleted still exists within Azure until the deletion process is complete.

`$RANDOM` is a bash function (not a constant) that returns a random signed 16-bit integer (from 0 through 32767). The `let` command is a built-in Bash command to evaluate

arithmetic expressions. Using the following command creates a sufficiently unique value for most purposes.

```
Azure CLI
```

```
let "randomIdentifier=$RANDOM*$RANDOM"
```

Working with spaces and quotation marks

Spaces are used for separating commands, options, and arguments. Use quote marks to tell the Bash shell to ignore all special characters, of which a white space is a special character. When the Bash shell sees the first quote mark, it ignores special characters until the closing quote mark. However, sometimes you want the Bash shell to parse certain special characters, such as dollar signs, back quotes, and backslashes. For this scenario, use double quotes.

The following commands use the [az group create](#) command to illustrate the use of single and double quote marks. These commands are used to handle spaces and evaluate special characters when working with variables and creating an object.

```
Azure CLI
```

```
resourceGroup='msdocs-learn-bash-$randomIdentifier'
echo $resourceGroup # The $ is ignored in the creation of the $resourceGroup
variable
resourceGroup="msdocs-learn-bash-$randomIdentifier"
echo $resourceGroup # The $randomIdentifier is evaluated when defining the
$resourceGroup variable
location="East US" # The space is ignored when defining the $location
variable
echo The value of the location variable is $location # The value of the
$location variable is evaluated
echo "The value of the location variable is $location" # The value of the
$location variable is evaluated
echo "The value of the location variable is \$location" # The value of the
$location variable is not evaluated
echo 'The value of the location variable is $location' # The value of the
$location variable is not evaluated
az group create --name $resourceGroup --location $location # Notice that the
space in the $location variable is not ignored and the command fails as it
treats the value after the space as a new command
az group create --name $resourceGroup --location "$location" # Notice that
the space in the $location variable is ignored and the location argument
accepts the entire string as the value
```

In the JSON dictionary output, review the properties of the resource group that was created.

Using If Then Else to determine if variable is null

To evaluate strings, use `!=` and to evaluate numbers use `-ne`. The following If Then Else statement evaluates whether the `$resourceGroup` variable has been set. If yes, it returns the value of the variable. If no, it sets the variable.

Azure CLI

```
if [ $resourceGroup != '' ]; then
    echo $resourceGroup
else
    resourceGroup="msdocs-learn-bash-$randomIdentifier"
fi
```

Using If Then to create or delete a resource group

The following script creates a new resource group only if one with the specified name doesn't already exist.

Azure CLI

```
if [ $(az group exists --name $resourceGroup) = false ]; then
    az group create --name $resourceGroup --location "$location"
else
    echo $resourceGroup
fi
```

The following script deletes an existing new resource group if one with the specified name already exists. You could use the `--no-wait` argument to return control without waiting for the command to complete. However, for this article, we want to wait for the resource group to be deleted before continuing. For more information on asynchronous operations, see [Tips for using the Azure CLI successfully - Asynchronous operations](#). We demonstrate the use of the `--no-wait` argument at the end of this article.

Azure CLI

```
if [ $(az group exists --name $resourceGroup) = true ]; then
    az group delete --name $resourceGroup -y # --no-wait
else
    echo The $resourceGroup resource group does not exist
fi
```

Using Grep to determine if a resource group exists, and create the resource group if it doesn't

The following command pipes the output of the `az group list` command to the `grep` command. If the specified resource group doesn't exist, the command creates the resource group using the previously defined variables.

Azure CLI

```
az group list --output tsv | grep $resourceGroup -q || az group create --name $resourceGroup --location "$location"
```

Using CASE statement to determine if a resource group exists, and create the resource group if it doesn't

The following CASE statement creates a new resource group only if one with the specified name doesn't already exist. If one with the specified name exists, the CASE statement echoes that the resource group exists.

Azure CLI

```
var=$(az group list --query "[? contains(name, '$resourceGroup')].name" --output tsv)
case $resourceGroup in
$var)
echo The $resourceGroup resource group already exists.;;
*)
az group create --name $resourceGroup --location "$location";;
esac
```

Using for loops and querying arrays

In this section of the article, we create a storage account and then use for loops to create blobs and containers. We also demonstrate querying JSON arrays and working with environment variables.

Create storage account

The following command uses the `az storage account create` command to create a storage account that we use when creating storage containers.

Azure CLI

```
storageAccount="learnbash$randomIdentifier"
az storage account create --name $storageAccount --location "$location" --
resource-group $resourceGroup --sku Standard_LRS --encryption-services blob
```

Get the storage account keys

The following commands use the [az storage account keys list](#) command to return storage account key values. We then store a key value in a variable for use when creating storage containers.

Azure CLI

```
az storage account keys list --resource-group $resourceGroup --account-name
$storageAccount --query "[].value" -o tsv # returns both storage account key
values

az storage account keys list --resource-group $resourceGroup --account-name
$storageAccount --query "[0].value" -o tsv # returns a single storage
account key value

accountKey=$(az storage account keys list --resource-group $resourceGroup --
account-name $storageAccount --query "[0].value" -o tsv)

echo $accountKey
```

Create storage container

We start by using the [az storage container create](#) to create a single storage container and then use the [az storage container list](#) to query the name of the created container.

Azure CLI

```
container="learningbash"
az storage container create --account-name $storageAccount --account-key
$accountKey --name $container

az storage container list --account-name $storageAccount --account-key
$accountKey --query [].name
```

Upload data to container

The following script creates three sample files using a for loop.

Azure CLI

```
for i in `seq 1 3`; do
    echo $randomIdentifier > container_size_sample_file_$i.txt
done
```

The following script uses the [az storage blob upload-batch](#) command to upload the blobs to the storage container.

Azure CLI

```
az storage blob upload-batch \
--pattern "container_size_sample_file_*.txt" \
--source . \
--destination $container \
--account-key $accountKey \
--account-name $storageAccount
```

The following script uses the [az storage blob list](#) command to list the blobs in the container.

Azure CLI

```
az storage blob list \
--container-name $container \
--account-key $accountKey \
--account-name $storageAccount \
--query "[].name"
```

The following script displays the total bytes in the storage container.

Azure CLI

```
bytes=`az storage blob list \
--container-name $container \
--account-key $accountKey \
--account-name $storageAccount \
--query "[*].[properties.contentLength]" \
--output tsv | paste -s -d+ | bc` 

echo "Total bytes in container: $bytes"
echo $bytes
```

Create many containers using loops

Next, we create multiple containers using a loop demonstrating a couple of ways to write the loop.

Azure CLI

```
for i in `seq 1 4`; do
az storage container create --account-name $storageAccount --account-key
$accountKey --name learnbash-$i
done

for value in {5..8}
for (( i=5; i<10; i++));
do
az storage container create --account-name $storageAccount --account-key
$accountKey --name learnbash-$i
done

az storage container list --account-name $storageAccount --account-key
$accountKey --query [].name
```

Use EXPORT to define environment variables

In the preceding storage container scripts, we specified the account name and account key with every command. Instead, you can store your authentication credentials using the corresponding environment variables: `AZURE_STORAGE_ACCOUNT` and `AZURE_STORAGE_KEY`. To perform this action, use `EXPORT`.

Azure CLI

```
export AZURE_STORAGE_ACCOUNT=$storageAccount
export AZURE_STORAGE_KEY=$accountKey
az storage container list # Uses the environment variables to display the
list of containers.
```

The following script creates a metadata string and then uses the `az storage container metadata update` command to update a container with that string, again using the environment variables.

Azure CLI

```
metadata="key=value pie=delicious" # Define metadata
az storage container metadata update \
--name $container \
--metadata $metadata # Update the metadata
az storage container metadata show \
--name $containerName # Show the metadata
```

The following command uses the `az storage container delete` command to delete a single named container and then delete multiple containers in a loop.

Azure CLI

```
az storage container delete \
--name $container
```

Get list of containers containing a specific prefix and store results into a variable.

Azure CLI

```
containerPrefix="learnbash"
containerList=$(az storage container list \
--query "[].name" \
--prefix $containerPrefix \
--output tsv)
```

Delete the list of containers in a loop using the `--prefix` argument.

Azure CLI

```
for row in $containerList
do
    tmpName=$(echo $row | sed -e 's/\r//g')
    az storage container delete \
    --name $tmpName
done
```

Error handling

To exit a script immediately if a command returns a nonzero status, run the following command:

Azure CLI

```
set -e
```

For more information about setting shell options and other help, run the following commands:

Azure CLI

```
help set
help help
```

Clean up resources

After you complete this article, delete the resource group and all resources within it. Use the `--no-wait` argument.

Azure CLI

```
if [ $(az group exists --name $resourceGroup) = true ]; then
    az group delete --name $resourceGroup -y --no-wait
else
    echo The $resourceGroup resource group does not exist
fi
```

See also

- [Tutorial: Learn to use the Azure CLI](#)
- [How to use the Azure CLI successfully](#)
- [How to query Azure CLI command output](#)
- Find Azure CLI samples and [published articles](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

-  [Open a documentation issue](#)
-  [Provide product feedback](#)

Considerations for running the Azure CLI in a PowerShell scripting language

Article • 06/27/2024

Azure CLI is a tool to manage Azure resources through Azure CLI reference commands that run in both a Bash and PowerShell scripting language. However, there are slight syntax differences in parameter formatting between scripting languages that can result in unexpected results. The purpose of this article is to help you resolve Azure CLI syntax errors when working in a PowerShell scripting language.

This article compares syntax differences of Azure CLI commands executed in the following scripting languages:

- Bash running in a Linux operating system using [Azure Cloud Shell](#).
- [PowerShell](#) running in a Linux operating system using Azure Cloud Shell.
- [Windows PowerShell](#) running in Windows 11 using the PowerShell 5 terminal.
- PowerShell running in a Windows 11 using the PowerShell 7 terminal.

If you're new to CLI, differentiating between a *tool* and a *scripting language* might be confusing. [How-to choose the right command-line tool](#) provides a good comparison.

Prerequisites

Read and learn

This article is intended for you to read and learn. However, if you want to run the examples, select the [Prepare your environments](#) tab to install the scripting languages used in this article.

ⓘ Important

When you have an Azure CLI script that is producing an error, consider how the scripting language you are working in is parsing the Azure CLI command syntax.

Pass spaces in Azure CLI parameters

In Azure CLI, when you need to pass a parameter value containing a space, there are quoting differences between operating systems and scripting languages. In this

example, use [az storage account list](#) and rename output columns with a word containing a space.

Bash in Linux

In this example, notice the single quote ('...') wrapper with embedded double quotes ("..."). This example also works in PowerShell in Linux.

Azure CLI

```
az storage account list --query '[].{\"SA Name\":name, \"Primary endpoint\":primaryEndpoints.blob}' --output table
```

If you want to add a filter, the syntax changes. Notice how this example wraps the `-query` parameter value in double quotes ("...") and uses a backslash (\) escape character. This script doesn't run in PowerShell.

Azure CLI

```
az storage account list --query "[?creationTime >='2024-02-01'].{\"SA Name\":name,\"Primary endpoint\":primaryEndpoints.blob}" --output table
```

If you just tried to run the filter syntax in a PowerShell scripting language, you received error message `argument --query: invalid jmespath_type value: "[? creationTime >=...]"`. However, in Bash within a Linux environment, your output is similar to this:

Output

SA Name	Primary Endpoint
msdocssa00000000	https://msdocssa00000000.blob.core.windows.net/

Pass parameters in a URL containing a query string

Question marks in URLs indicate the end of the URL and the beginning of a query string. Here's an example that opens step 3 in [Learn to use the Azure CLI](#):

<https://learn.microsoft.com/en-us/cli/azure/account?view=azure-cli-2020-09-01-hybrid>.

The `?view=azure-cli-2020-09-01-hybrid` results in the desired version of the Azure CLI reference content.

When you execute Azure CLI commands in a PowerShell scripting language, PowerShell allows question marks to be part of a variable name. This might create confusion in Azure CLI parameter values.

Here's an example from the [Use the Azure REST API](#) article:

Bash

Notice how `$containerRegistryName?api-version` concatenates together without error in Bash.

Azure CLI

```
# Script for a Bash scripting language

# Variable block
let "randomIdentifier=$RANDOM*$RANDOM"
subscriptionId="00000000-0000-0000-0000-000000000000"
resourceGroup="msdocs-app-rg$randomIdentifier"
containerRegistryName="msdocscr$randomIdentifier"

# prior to this GET example, the resource group and container registry
# were created in the article.

az rest --method get --url
https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName?api-version=2023-01-01-preview
```

Pass parameters containing the ampersand symbol

If you have a scenario where you need to pass an ampersand in a parameter value, be aware that the ampersand (`&`) symbol is interpreted by PowerShell. You can see this happen using the `--debug` parameter:

Azure CLI

```
az "a&b" --debug

# output
```

```
'a' is misspelled or not recognized by the system.  
'b' is not recognized as an internal or external command
```

However, if you use this same test to add a tag to a resource group, the ampersand in the tag value doesn't cause an error.

Azure CLI

```
az group create --location eastus2 --name "msdocs-rg-test"  
az group update --name "msdocs-rg-test" --tags "company name=Contoso & Sons"  
  
# output  
{  
  "id": "/subscriptions/3618afcd-ea52-4ceb-bb46-  
53bb962d4e0b/resourceGroups/msdocs-rg-test",  
  "location": "eastus2",  
  "managedBy": null,  
  "name": "msdocs-rg-test",  
  "properties": {  
    "provisioningState": "Succeeded"  
  },  
  "tags": {  
    "company name": "Contoso & Sons"  
  },  
  "type": "Microsoft.Resources/resourceGroups"  
}
```

If you have a scenario where the ampersand in a parameter value is causing an error, here are some solutions:

Azure CLI

```
# When quoted by single quotes ('), double quotes ("") are preserved by  
PowerShell and sent  
# to Command Prompt, so that ampersand (&) is treated as a literal character  
> az '"a&b"' --debug  
Command arguments: ['a&b', '--debug']  
  
# Escape double quotes (") with backticks (`) as required by PowerShell  
> az `"\&b`"" --debug  
Command arguments: ['a&b', '--debug']  
  
# Escape double quotes (") by repeating them  
> az """a&b""" --debug  
Command arguments: ['a&b', '--debug']  
  
# With a whitespace in the argument, double quotes (") are preserved by  
PowerShell and  
# sent to Command Prompt  
> az "a&b " --debug  
Command arguments: ['a&b ', '--debug']
```

```
# Use --% to stop PowerShell from parsing the argument
> az --% "a&b" --debug
Command arguments: ['a&b', '--debug']
```

Pass parameters containing an *at* (@) symbol

There are special characters of PowerShell, such as the *at* (@) symbol which is a [splatting operator](#) in PowerShell. Add a backtick ` before the special character to escape it. You can also enclose the value in single (') or double (") quotes.

The following three examples will work in PowerShell:

- parameterName `@parameters.json
- parameterName '@parameters.json'
- parameterName "@parameters.json"

This example will not work in PowerShell:

- parameterName @parameters.json

Here's another example in the `az ad app create` command: Notice the double quotes ("...") around the JSON file name needed in a PowerShell scripting language.

Azure CLI

```
# Script for a PowerShell scripting language

az ad app create --display-name myTestAppName ` 
    --is-fallback-public-client ` 
    --required-resource-accesses "@manifest.json"
```

Pass parameters containing JSON

For complex arguments like a JSON string, the best practice is to use Azure CLI's `@<file>` convention to load from a file to bypass the shell's interpretation. However, here are some additional JSON parameter format values that are correct in PowerShell:

Azure CLI

```
# Correct
> az '{\"key\": \"value\"}' --debug
Command arguments: ['{"key": "value"}', '--debug']
```

```
> az "{\"key\": \"value\"}" --debug
Command arguments: ['{"key": "value"}', '--debug']

> az "\"key\": \"value\"" --debug
Command arguments: ['{"key": "value"}', '--debug']

> az --% "{\"key\": \"value\"}" --debug
Command arguments: ['{"key": "value"}', '--debug']

# Wrong!
> az '{"key": "value"}' --debug
Command arguments: ['{key: value}', '--debug']
```

Pass parameters containing key:value pairs

Some Azure CLI parameter values, such as Azure resource tags, require key:value pairs. If your `key` or `value` contains a space or special character, the Bash and PowerShell syntax aren't always the same.

See [Create tags to practice quoting differences](#) in the [Learn to use the Azure CLI](#) tutorial. This tutorial step gives examples for Bash, PowerShell, and Cmd for the following key:value pair scenarios:

- spaces
- empty values
- special characters
- variables

Stop-parsing symbol

The stop-parsing symbol (`--%`), introduced in PowerShell 3.0, directs PowerShell to refrain from interpreting input as PowerShell commands or expressions. When it encounters a stop-parsing symbol, PowerShell treats the remaining characters in the line as a literal.

Azure CLI

```
az --% vm create --name xxx
```

Error handling for Azure CLI in PowerShell

You can run Azure CLI commands in PowerShell, as described in [Choose the right Azure command-line tool](#). If you do, be sure you understand Azure CLI error handling in

PowerShell. In particular, Azure CLI doesn't create exceptions for PowerShell to catch.

An alternative is to use the `$?` automatic variable. This variable contains the status of the most recent command. If the previous command fails, `$?` has the value of `$False`. For more information, see [about_Automatic_Variables](#).

The following example shows how this automatic variable can work for error handling:

```
Azure CLI

# Script for a PowerShell scripting language

az group create --name MyResourceGroup
if ($? -eq $false) {
    Write-Error "Error creating resource group."
}
```

The `az` command fails because it's missing the required `--location` parameter. The conditional statement finds that `$?` is false and writes an error.

If you want to use the `try` and `catch` keywords, you can use `throw` to create an exception for the `try` block to catch:

```
Azure CLI

# Script for a PowerShell scripting language

$ErrorActionPreference = "Stop"
try {
    az group create --name MyResourceGroup
    if ($? -eq $false) {
        throw 'Group create failed.'
    }
}
catch {
    Write-Error "Error creating the resource group."
}
$ErrorActionPreference = "Continue"
```

By default, PowerShell catches only terminating errors. This example sets the `$ErrorActionPreference` global variable to `Stop` so PowerShell can handle the error.

The conditional statement tests the `$?` variable to see if the previous command failed. If so, the `throw` keyword creates an exception to catch. The `catch` block can be used to write an error message or handle the error.

The example restores `$ErrorActionPreference` to its default value.

For more information about PowerShell error handling, see [Everything you wanted to know about exceptions](#).

Enable Tab Completion in PowerShell

Tab completion, also known as "Azure CLI completers", provides completion on inputs to provide hints, enable discovery and speed up input entry. Command names, command group names, parameters and certain parameter values can be automatically inserted into the command line by pressing the `Tab` key.

Tab completion is enabled by default in Azure Cloud Shell and in most Linux distributions. Starting in Azure CLI version 2.49, you can enable tab completion for the Azure CLI in PowerShell. Follow these steps:

1. Create or edit the profile stored in the variable `$PROFILE`. The simplest way is to run `notepad $PROFILE` in PowerShell. For more information, see [How to create your profile](#) and [Profiles and execution policy](#).
2. Add the following code to your PowerShell profile:

```
PowerShell

Register-ArgumentCompleter -Native -CommandName az -ScriptBlock {
    param($commandName, $wordToComplete, $cursorPosition)
    $completion_file = New-TemporaryFile
    $env:_ARGCOMPLETE_USE_TEMPFILES = 1
    $env:_ARGCOMPLETE_STDOUT_FILENAME = $completion_file
    $env:COMP_LINE = $wordToComplete
    $env:COMP_POINT = $cursorPosition
    $env:_ARGCOMPLETE = 1
    $env:_ARGCOMPLETE_SUPPRESS_SPACE = 0
    $env:_ARGCOMPLETE_IFS = "`n"
    $env:_ARGCOMPLETE_SHELL = 'powershell'
    az 2>&1 | Out-Null
    Get-Content $completion_file | Sort-Object | ForEach-Object {
        [System.Management.Automation.CompletionResult]::new($_, $_,
        "ParameterValue", $_)
    }
    Remove-Item $completion_file, Env:_ARGCOMPLETE_STDOUT_FILENAME,
    Env:_ARGCOMPLETE_USE_TEMPFILES, Env:COMP_LINE, Env:COMP_POINT,
    Env:_ARGCOMPLETE, Env:_ARGCOMPLETE_SUPPRESS_SPACE,
    Env:_ARGCOMPLETE_IFS, Env:_ARGCOMPLETE_SHELL
}
```

3. To display all available options in the menu, add `Set-PSReadlineKeyHandler -Key Tab -Function MenuComplete` to your PowerShell profile.

See also

- Azure CLI engineering notes on [Quoting issues with PowerShell ↗](#)
- Compare syntax of Bash, PowerShell, and Cmd in these articles:
 - [Quoting differences between scripting languages](#)
 - [Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#) tutorial
 - [Query Azure CLI command output using a JMESPath query](#)

 **Collaborate with us on GitHub**

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project.
Select a link to provide feedback:

- [Open a documentation issue](#)
- [Provide product feedback](#)

Use Azure CLI to manage sensitive information

Article • 05/24/2024

When you manage Azure resources, the output of an Azure CLI command might expose sensitive information that must be protected. For example, keys, passwords and connection strings can be created by Azure CLI commands and displayed in a terminal window. The output for some commands can also be stored in log files. This is often the case when working with GitHub actions and other DevOps runners.

It's critical to protect this information! If acquired publicly from environments with lesser permissions, the exposure of secrets can cause serious damage and lead to a loss of trust in your company's products and services. To help you to protect sensitive information, the Azure CLI detects secrets in the output of some reference commands, and displays a warning message when a secret is identified.

Set secrets warning configuration

Beginning in [Azure CLI 2.61](#), a warning message is displayed when reference commands result in the output of sensitive information.

Sensitive information warnings are **enabled** by default. Turn sensitive information warnings off by setting the `clients.show_secrets_warning` configuration property to `no`.

Azure CLI

```
az config set clients.show_secrets_warning=no
```

Considerations

The purpose of the warning message is to decrease the unintentional exposure of secrets, but these messages might require you to make changes in existing scripts.

Important

The new warning messages are sent to Standard Error (STDERR), not Standard Out (STDOUT). Therefore, if you are running an Azure CLI command that results in

sensitive information output, you might need to trap for the warning message, or turn warnings off.

For example, in Azure DevOps Services [pipelines](#), if the `failOnStderr` parameter is set to `True` of the Bash v3 task, the warning message halts the pipeline. Consider enabling the `show_secrets_warning` message to identify if any secrets are exposed in your pipelines, and then take remediation actions.

See also

- [Azure CLI configuration](#)
- [Azure CLI output format of "none"](#)
- [Microsoft guidance regarding credentials leaked to GitHub Actions Logs through Azure CLI](#)

Azure CLI configuration

Article • 06/05/2024

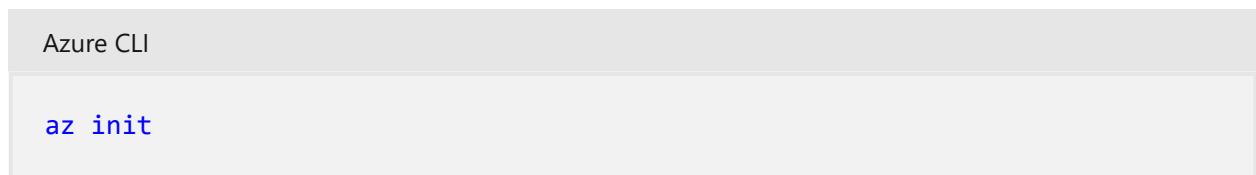
The Azure CLI allows for user configuration for settings such as logging, data collection, and default argument values. The CLI offers a convenience command for managing some defaults, `az config`, and an interactive option through `az init`. Other values can be set in a configuration file or with environment variables. This article provides further information on these user configuration settings and how to configure the Azure CLI.

Configuration values used by the CLI are evaluated in the following precedence, with items higher on the list taking priority.

1. Command-line parameters
2. Environment variables
3. Values in the configuration file set with `az config` or `az init`

Configure settings using `az init`

The easiest way to set Azure CLI configurations is to use the interactive `az init` command. Choose a common configuration like "interaction" or "automation," or opt to walk through individual configurations. What is especially helpful with this approach is `az init` provides reasons why you might want to choose one configuration option over another.



Configure settings using `az config`

You can set defaults for the CLI with the `az config set` command. This command takes a space-separated list of `key=value` pairs as an argument. The provided values are used by the Azure CLI in place of required arguments.

The following table contains a list of available configuration keys.

[] Expand table

Name	Description
defaults.group	The default resource group to use for all commands.
defaults.location	The default location to use for all commands.
defaults.web	The default app name to use for <code>az webapp</code> commands.
defaults.vm	The default VM name to use for <code>az vm</code> commands.
defaults.vmss	The default virtual machine scale set (VMSS) name to use for <code>az vmss</code> commands.
defaults.acr	The default container registry name to use for <code>az acr</code> commands.

As an example, here's how you would set the default resource group and location for all commands.

Azure CLI

```
az config set defaults.location=westus2 defaults.group=MyResourceGroup
```

The following command turns off the survey links while executing the Azure CLI commands:

Azure CLI

```
az config set output.show_survey_link=no
```

CLI configuration file

The CLI configuration file contains other settings that are used for managing CLI behavior. The configuration file itself is located at `$AZURE_CONFIG_DIR/config`. The default value of `AZURE_CONFIG_DIR` is `$HOME/.azure` on Linux and macOS, and `%USERPROFILE%\azure` on Windows.

Configuration files are written in the INI file format. Section headers define the file format, followed by a list of key-value entries.

- Section headers are written as `[section-name]`. Section names are case-sensitive.
- Entries are written as `key=value`. Key names aren't case-sensitive.
- Comments are any line that begins with a `#` or `;`. Inline comments aren't allowed.

Booleans are case-insensitive. The following values represent booleans:

- True: `1`, `yes`, `true`, `on`
- False: `0`, `no`, `false`, `off`

Here's an example of a CLI configuration file that disables any confirmation prompts and sets up logging to the `/var/log/azure` directory.

```
ini

[core]
disable_confirm_prompt=Yes

[logging]
enable_log_file=yes
log_dir=/var/log/azure
```

See the next section for details on all of the available configuration values and what they mean. For the full details on the INI file format, see the [Python documentation on INI](#).

CLI configuration values and environment variables

The following table contains all of the sections and option names that can be placed in a configuration file. Their corresponding environment variables are set as

`AZURE_{section}_{name}`, in all caps. For example, `output` default for `core` is set in the `AZURE_CORE_OUTPUT` variable, the `storage_account` default for `batchai` is set in the `AZURE_BATCHAI_STORAGE_ACCOUNT` variable, and the default `location` is set in the `AZURE_DEFAULTS_LOCATION` variable.

When you provide a default value, that argument is no longer required by any command. Instead, the default value is used.

[+] Expand table

Section	Name	Type	Description
core	output	string	The default output format. Allowed values: <code>json</code> (default), <code>jsonc</code> , <code>yaml</code> , <code>yamlc</code> , <code>table</code> , <code>tsv</code> , <code>none</code> . For more information, see Output formats for Azure CLI commands .
	disable_confirm_prompt	boolean	Turn confirmation prompts on/off.
	display_region_identified	boolean	Azure customers can choose to deploy resources in many different regions. In some cases,

Section	Name	Type	Description
			customers may be able to reduce costs by selecting nearby regions offering the same services. If a nearby region is identified, a message will display the region to select for future deployments. This setting controls if the message is displayed.
	collect_telemetry	boolean	Allow Microsoft to collect anonymous data on the usage of the CLI. For privacy information, see the Azure CLI MIT license .
	only_show_errors	boolean	Only show errors during command invocation. In other words, only errors are written to <code>stderr</code> . It suppresses warnings from preview, deprecated and experimental commands. It's also available for individual commands with the <code>--only-show-errors</code> parameter.
	enable_broker_on_windows	boolean	Use Web Account Manager (WAM) to authenticate to Azure through the <code>az login</code> command.
	login_experience_v2	boolean	Turn the <code>az login</code> subscription selector on/off.
	no_color	boolean	Disable color. Originally colored messages are prefixed with <code>DEBUG</code> , <code>INFO</code> , <code>WARNING</code> and <code>ERROR</code> . This boolean bypasses the issue of a third-party library where the terminal's color can't revert back after a <code>stdout</code> redirection.
clients	show_secrets_warning	boolean	Turn the warning for sensitive information output on/off.
logging	enable_log_file	boolean	Turn logging on/off.
	log_dir	string	The directory to write logs to. By default this value is <code> \${AZURE_CONFIG_DIR}/logs*</code> .
defaults	group	string	The default resource group to use for all commands.
	location	string	The default location to use for all commands.
	web	string	The default app name to use for <code>az webapp</code> commands.
	vm	string	The default VM name to use for <code>az vm</code> commands.

Section	Name	Type	Description
	vmss	string	The default virtual machine scale set (VMSS) name to use for <code>az vmss</code> commands.
	acr	string	The default container registry name to use for <code>az acr</code> commands.
storage	account	string	The default storage account name (for example, <code>mystorageaccount</code> in <code>https://mystorageaccount.blob.core.windows.net</code>) to use for <code>az storage</code> data-plane commands (for example, <code>az storage container list</code>).
	key	string	The default access key to use for <code>az storage</code> data-plane commands.
	sas_token	string	The default SAS token to use for <code>az storage</code> data-plane commands.
	connection_string	string	The default connection string to use for <code>az storage</code> data-plane commands.
batchai	storage_account	string	The default storage account to use for <code>az batchai</code> commands.
	storage_key	string	The default storage key to use for <code>az batchai</code> commands.
batch	account	string	The default Azure Batch account name to use for <code>az batch</code> commands.
	access_key	string	The default access key to use for <code>az batch</code> commands. Only used with <code>aad</code> authorization.
	endpoint	string	The default endpoint to connect to for <code>az batch</code> commands.
	auth_mode	string	The authorization mode to use for <code>az batch</code> commands. Allowed values: <code>shared_key</code> , <code>aad</code> .
cloud	name	string	The default cloud for all <code>az</code> commands. Allowed values: <code>AzureCloud</code> (default), <code>AzureChinaCloud</code> , <code>AzureUSGovernment</code> . To change clouds, you can use the <code>az cloud set -name</code> command. For an example, see Manage Clouds with the Azure CLI .
extension	use_dynamic_install	string	Install an extension if it's not added yet when running a command from it. Allowed values: <code>no</code> (default), <code>yes_prompt</code> , <code>yes_without_prompt</code> .

Section	Name	Type	Description
	run_after_dynamic_install	boolean	Continue to run the command when an extension is dynamically installed for it. Default is <code>False</code> .
	index_url	string	URL of private extension index file following the format in index.json . Once specified, executing <code>az extension add --name <extension-name></code> uses that file to find the extension to add.

Note

You may see other values in your configuration file, but these are managed directly through CLI commands, including `az config`. The ones listed in the table previously are the only values you should change yourself.

How to manage Azure subscriptions with the Azure CLI

Article • 08/27/2024

The Azure CLI helps you manage your Azure subscription, create management groups, and lock subscriptions. You might have multiple subscriptions within Azure. You can be part of more than one organization or your organization might divide access to certain resources across groupings. The Azure CLI supports selecting a subscription both globally and per command.

For detailed information on subscriptions, billing, and cost management, see the [billing and cost management documentation](#).

Terminology

A *tenant* is an instance of Microsoft Entra ID in which information about a single organization resides. A *multi-tenant organization* is an organization that has more than one instance of Microsoft Entra ID. A tenant has one or more *subscriptions* and *users*.

Users are those accounts that sign in to Azure to create, manage, and use resources. A user may have access to multiple *tenants* and *subscriptions*.

Subscriptions are the agreements with Microsoft to use cloud services, including Azure. Every resource is associated with a subscription. Subscriptions contain resource groups.

An Azure *resource group* is a container that holds related resources for an Azure solution. To learn how to manage resource groups within your subscription, see [How to manage Azure resource groups with the Azure CLI](#)

Get the active tenant

Use [az account tenant list](#) or [az account show](#) to get the active tenant ID.

```
Azure CLI
az account tenant list
az account show
```

Change the active tenant

To switch tenants, you have two options.

- [Change the active subscription.](#)
- Sign in as a user within the desired tenant. Use [az login](#) to change the active tenant and update the subscription list to which you belong.

Azure CLI

```
# sign in as a different user
az login --user <myAlias@myCompany.com> --password <myPassword>

# sign in with a different tenant
az login --tenant <myTenantID>
```

If your organization requires multi-factor authentication, you may receive this error when using `az login --user`:

Output

```
Due to a configuration change made by your administrator, or because
you moved to a new location, you must use multi-factor authentication
to access...
```

Using the alternative `az login --tenant` command prompts you to open an HTTPS page and enter the code provided. You can then use multi-factor authentication and successfully sign in. To learn more about sign in options with the Azure CLI, see [Sign in with the Azure CLI](#).

Get subscription information

Most Azure CLI commands act within a subscription. You can specify which subscription to work in by using the `--subscription` parameter in your command. If you don't specify a subscription, the command uses your current, active subscription.

To see the subscription you're currently using or to get a list of available subscriptions, run the [az account show](#) or [az account list](#) command. Go to [Learn to use Bash with the Azure CLI](#) to see more examples of ways to use these commands.

Here are examples showing how to get subscription information:

Azure CLI

```
# get the current default subscription using show
az account show --output table

# get the current default subscription using list
az account list --query "[?isDefault]"

# get a subscription that contains search words or phrases
az account list --query "[?contains(name,'search phrase')].
{SubscriptionName:name, SubscriptionID:id, TenantID:tenantId}" --output
table
```

You can also store subscription information in a variable for use within a script.

Bash

Azure CLI

```
# store the default subscription in a variable
subscriptionId=$(az account list --query "[?isDefault].id" --output
tsv)"
echo $subscriptionId

# store a subscription of certain name in a variable
subscriptionId=$(az account list --query "[?name=='my case sensitive
subscription full name'].id" --output tsv)"
echo $subscriptionId
```

Tip

The `--output` parameter is a global parameter, available for all commands. The `table` value presents output in a friendly format. For more information, see [Output formats for Azure CLI commands](#).

Change the active subscription

Azure subscriptions have both a name and an ID. You can switch to a different subscription using `az account set` specifying the desired subscription ID or name.

Azure CLI

```
# change the active subscription using the subscription name
az account set --subscription "My Demos"
```

```
# change the active subscription using the subscription ID  
az account set --subscription "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
```

You can also change your subscription using a variable. Here is an example:

Bash

Azure CLI

```
# change the active subscription using a variable  
subscriptionId=$(az account list --query "[?name=='my case sensitive  
subscription full name'].id" --output tsv)"  
az account set --subscription $subscriptionId
```

If you change to a subscription that is in a different tenant, you will also be changing the active tenant. To learn how to add a new subscription to your Microsoft Entra tenant, see [Associate or add an Azure subscription to your Microsoft Entra tenant](#).

If you received a "The subscription of ... doesn't exist..." error, see [Troubleshooting](#) for possible solutions.

Clear your subscription cache

To update your subscription list, use the `az account clear` command. You will need to sign in again to see an updated list.

Azure

```
az account clear  
  
az login
```

Clearing your subscription cache is not technically the same process as logging out of Azure. However, when you clear your subscription cache, you cannot run Azure CLI commands, including `az account set`, until you sign in again.

Create Azure management groups

Azure management groups contain subscriptions. Management groups provide a way to manage access, policies, and compliance for those subscriptions. For more information, see [What are Azure management groups](#).

Use the [az account management-group](#) commands to create and manage Azure Management Groups.

You can create a management group for several of your subscriptions by using the [az account management-group create](#) command:

Azure CLI

```
az account management-group create --name Contoso01
```

To see all your management groups, use the [az account management-group list](#) command:

Azure CLI

```
az account management-group list
```

Add subscriptions to your new group by using the [az account management-group subscription add](#) command:

Azure CLI

```
az account management-group subscription add --name Contoso01 --subscription "My Demos"  
az account management-group subscription add --name Contoso01 --subscription "My Second Demos"
```

To remove a subscription, use the [az account management-group subscription remove](#) command:

Azure CLI

```
az account management-group subscription remove --name Contoso01 --subscription "My Demos"
```

To remove a management group, run the [az account management-group delete](#) command:

Azure CLI

```
az account management-group delete --name Contoso01
```

Removing a subscription or deleting a management group doesn't delete or deactivate a subscription.

Set an Azure subscription lock

As an administrator, you may need to lock a subscription to prevent users from deleting or modifying it. For more information, see [Lock resources to prevent unexpected changes](#).

In Azure CLI, use the `az account lock` commands. For instance, the `az account lock create` command can prevent users from deleting a subscription:

Azure CLI

```
az account lock create --name "Cannot delete subscription" --lock-type  
CanNotDelete
```

ⓘ Note

You need to have `contributor` permissions on a subscription to create or change locks.

To see the current locks on your subscription, use the `az account lock list` command:

Azure CLI

```
az account lock list --output table
```

If you make an account read-only, the result resembles assigning permissions of the Reader role to all users. To learn about setting permissions for individual users and roles, see [Add or remove Azure role assignments using Azure CLI](#).

To see details for a lock, use the `az account lock show` command:

Azure CLI

```
az account lock show --name "Cannot delete subscription"
```

You can remove a lock by using the `az account lock delete` command:

Azure CLI

```
az account lock delete --name "Cannot delete subscription"
```

Troubleshooting

The subscription doesn't exist

In addition to a typographical error, you can receive this error when there is a permissions timing issue. For example, if you have been given permissions to a new subscriptions *while your current terminal window is open*, this error can occur. The solution is to either close and reopen your terminal window, or use `az logout` then `az login` to refresh your available subscriptions list.

Here is a script to help you find and change a subscription.

```
Azure CLI

# See what subscription you are currently using.
az account show

# Get a list of available subscriptions.
az account list --output table

# If the subscription you are seeking is not in the list
#   close and reopen your terminal window,
#   or logout and then sign in again.
az logout
az login

# You can also clear your cache to refresh the
#   available subscription list
az account clear
az login

# Did your available subscription list change?
az account list --output table

# If the subscription you are seeking is still not in the list,
#   contact your system administrator. You cannot change your
#   subscription to an ID that is not in the list.

# If the subscription you are seeking is now in the list,
#   change your subscription.
az account set --subscription 00000000-0000-0000-0000-000000000000
```

See also

- Associate or add an Azure subscription to your Microsoft Entra tenant
- Manage Azure resource groups

How to manage Azure resource groups with the Azure CLI

Article • 04/08/2024

An Azure resource group is a container that holds related resources for an Azure solution. A resource group might contain storage, virtual machines, apps, dashboards, services, or almost anything you deal with in Azure.

The Azure Command-Line Interface (CLI) allows you to create, persist, and set default Azure resource groups. The CLI will also allow you to clean up resources after creating them.

Azure Region Identification

Azure customers can choose to deploy resources in many different regions. In some cases, customers may be able to reduce costs by selecting nearby regions offering the same services. If a nearby region is identified, a message will display the region to select for future deployments.

In the following example, the `az config` command is used to disable the region recommendation message:

```
Azure CLI  
az config set core.display_region_identified=no
```

For more information about Azure regions, see [Choose the right Azure region for you](#).

Create a resource group

To create a resource group, use the `az group create` command:

```
Azure CLI  
az group create --name MyResourceGroup --location eastus
```

A resource group belongs to a single location. To see all the locations supported in your current subscription, run the `az account list-locations` command:

```
Azure CLI
```

```
az account list-locations
```

To see all the resource groups for your current subscription, use the [az group list](#) command:

Azure CLI

```
az group list --output table
```

💡 Tip

The `--output` parameter is a global parameter, available for all commands. The `table` value presents output in a friendly format. For more information, see [Output formats for Azure CLI commands](#).

When you create a resource, you create it in a resource group. The following example shows a storage account created by using the [az storage account create](#) command:

Azure CLI

```
az storage account create --resource-group MyResourceGroup --name storage134  
--location eastus --sku Standard_LRS
```

To remove a resource group, run the [az group delete](#) command:

Azure CLI

```
az group delete --name MyResourceGroup
```

When you remove a resource group, you delete all the resources that belong to it. You can't undo this action. If you try any of the commands in this article, deleting the resource groups you create cleans up your account.

Set a default resource group

You can set a default resource group for all the commands that you run from your local Azure CLI or from Azure Cloud Shell. Azure CLI stores this configuration locally in a `config` file. To see your current configuration, run the [az config get](#) command:

Azure CLI

```
az config get
```

The result shows default resource groups and other default values. If you're using Azure CLI for the first time, the results might be empty.

To set a default resource group for your Azure CLI installation, run the [az config set](#) command:

```
Azure CLI
```

```
az config set defaults.group=MyResourceGroup
```

The command sets a value for a specified key, in this case `defaults.group`. For available configuration options, see [Azure CLI configuration](#).

ⓘ Note

The [az config set](#) command does not validate the existence of the resource group you enter. The command simply stores the key-value pair.

After you run the command, the following two commands would give you the same result:

```
Azure CLI
```

```
az storage account create --resource-group MyResourceGroup --name storage01  
--location eastus --sku Standard_LRS  
az storage account create --name storage01 --location eastus --sku  
Standard_LRS
```

A resource group belongs to a subscription. If your organization has more than one subscription, you need to set that subscription before working with a resource group in the subscription. If the default value of a resource group doesn't belong to your current subscription, an error results. For more information about multiple subscriptions, see [Use multiple Azure subscriptions](#).

You don't have to reset the default to use other resource groups. Instead, specify the resource group:

```
Azure CLI
```

```
az group create --name OtherResourceGroup --location eastus  
az storage account create --resource-group StorageGroups --name storage03 -
```

```
-location westus --sku Standard_LRS
```

The default value is for you only. It doesn't affect other users or changes you make through the Azure portal.

If you're using persisted parameter values, as described in this article, those values take precedence over defaults set in the *config* file.

Set a resource group lock

As an administrator, you may need to lock a resource group to prevent users from deleting or modifying it. For more information, see [Lock resources to prevent unexpected changes](#).

In Azure CLI, use the [az group lock](#) commands. For instance, the [az account lock create](#) command can prevent users from deleting a resource group:

Azure CLI

```
az group lock create --name "Cannot delete resource group" --lock-type  
CanNotDelete
```

ⓘ Note

You need to have `contributor` permissions on a resource group to create or change locks.

To see the current locks on your resource group, use the [az group lock list](#) command:

Azure CLI

```
az group lock list --output table
```

Clean up resources

If you tried any of the commands in this article, you can remove any resources you created by using the [az group delete](#) command:

Azure CLI

```
az group delete --name MyResourceGroup  
az group delete --name OtherResourceGroup
```

```
az group delete --name StorageGroups
```

This command removes the group and all the resources that it contains at once.

See also

- [Azure CLI configuration](#)
- [Manage Azure subscriptions](#)

How to work with Azure identities using the Azure CLI

Article • 09/02/2024

There are many Azure CLI reference commands that help you to work with Azure identities. This article provides a list of Azure CLI commands and the quickstart, how-to guide or tutorial that shows the use of the command.

ⓘ Note

Not all Azure CLI commands containing the word "identity" are about Microsoft Entra ID *managed* identities. This article is an index of Azure CLI commands to help you manage any type of Azure identity.

az acr

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az acr identity	Troubleshoot a customer-managed key

az aks

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az aks pod-identity	Migrate from pod managed-identity to workload identity
	Use Microsoft Entra pod-managed identities in Azure Kubernetes Service (Preview)
az aks pod-identity exception	Troubleshoot Azure Kubernetes Service backup and restore
	Use Microsoft Entra pod-managed identities in Azure Kubernetes Service (Preview)

az appconfig

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az appconfig identity	Use customer-managed keys to encrypt your App Configuration data
	How to use managed identities for Azure App Configuration

az backup

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az backup vault identity	Save and manage MARS agent passphrase securely in Azure Key Vault
	Encrypt backup data by using customer-managed keys

az communication

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az communication identity	How to use Managed Identity with Azure Communication Services
az communication identity token	Quickstart: Join a room call
	Pre-Call diagnostic
	Adding visual effects to a video call

az containerapp

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az containerapp identity	Managed identities in Azure Container Apps Deploy to Azure Container Apps from Azure Pipelines Deploy to Azure Container Apps with GitHub Actions Migrate a Python application to use passwordless connections with Azure SQL Database Migrate an application to use passwordless connections with Azure Blob Storage Migrate an application to use passwordless connections with Azure Database for PostgreSQL Migrate an application to use passwordless connections with Azure Service Bus Tutorial: Use code interpreter sessions in LangChain with Azure Container Apps Migrate an application to use passwordless connections with Azure Event Hubs Tutorial: Use code interpreter sessions in Semantic Kernel with Azure Container Apps

az cosmosdb

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az cosmosdb identity	Configure managed identities with Microsoft Entra ID for your Azure Cosmos DB account Access Azure Key Vault from Azure Cosmos DB using a managed identity Create and manage container copy jobs in Azure Cosmos DB (Preview) Configure customer-managed keys for your Azure Cosmos DB account with Azure Managed HSM Key Vault

az dt

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az dt identity	Set up an Azure Digital Twins instance and authentication (CLI)

az functionapp

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az functionapp identity	Create your first containerized functions on Azure Container Apps
	Enable linting and analysis for API governance in your API center
	Tutorial: Build out an end-to-end solution
	Ingest IoT Hub telemetry into Azure Digital Twins
	Set up twin-to-twin event handling
	Integrate Azure Digital Twins with Azure Time Series Insights

az identity

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az identity	How to use managed identities for App Service and Azure Functions
	Manage user-assigned managed identities
	Use Key Vault references as app settings in Azure App Service and Azure Functions
	Transfer an Azure subscription to a different Microsoft Entra directory
	Configure a custom container for Azure App Service
	Deploy and configure workload identity on an Azure Kubernetes Service (AKS) cluster
	Migrate custom software to Azure App Service using a custom container
	Provision and publish a bot

Reference subgroup	Azure CLI article showing reference use
	Configure a user-assigned managed identity to trust an external identity provider
	Assign Azure roles using Azure CLI
az identity federated-credential	Deploy and configure workload identity on an Azure Kubernetes Service (AKS) cluster
	Configure a user-assigned managed identity to trust an external identity provider
	Tutorial: Deploy applications using GitOps with Flux v2
	Migrate from pod managed-identity to workload identity
	Quickstart: Deploy Application Gateway for Containers ALB Controller
	Integrate KEDA with your Azure Kubernetes Service cluster
	Send Prometheus data to Azure Monitor by using Microsoft Entra Workload ID (preview) authentication
	Deploy a highly available PostgreSQL database on AKS
	Configure and deploy a Valkey cluster on Azure Kubernetes Service (AKS)
	Securely scale your applications using the KEDA add-on and workload identity on Azure Kubernetes Service (AKS)

az image

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az image builder identity	Troubleshoot Azure VM Image Builder

az iot

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az iot central app identity	Export IoT data to Blob Storage

Reference subgroup	Azure CLI article showing reference use
	Export IoT data to Azure Data Explorer
	Export IoT data to Event Hubs
	Manage and monitor IoT Central applications
	Export IoT data to Service Bus
az iot hub device-identity	Create and provision an IoT Edge device on Linux using symmetric keys
	Quickstart: Deploy your first IoT Edge module to a virtual Linux device
	Install and run the Spatial Analysis container (preview)
	Tutorial: Send device data to Azure Storage using IoT Hub message routing
	Quickstart: Deploy your first IoT Edge module to a Windows device
	Quickstart: Send telemetry from a device to an IoT hub and monitor it with the Azure CLI
	Tutorial - Use MQTT to develop an IoT device client without using a device SDK
	Tutorial: Build out an end-to-end solution
	Create and provision an IoT Edge for Linux on Windows device using symmetric keys
	Tutorial: Configure your devices from a back-end service
az iot hub device-identity connection-string	Create and provision an IoT Edge device on Linux using symmetric keys
	Quickstart: Deploy your first IoT Edge module to a virtual Linux device
	Install and run the Spatial Analysis container (preview)
	Quickstart: Deploy your first IoT Edge module to a Windows device
	Tutorial: Build out an end-to-end solution
	Create and provision an IoT Edge for Linux on Windows device using symmetric keys

Reference subgroup	Azure CLI article showing reference use
	Use Visual Studio 2022 to develop and debug modules for Azure IoT Edge
	Tutorial: Configure your devices from a back-end service
	Tutorial: Use a simulated device to test connectivity with your IoT hub
	Tutorial: Set up and use metrics and logs with an IoT hub
<code>az iot hub identity</code>	Configure IoT Hub file uploads using Azure CLI
<code>az iot hub module-identity</code>	Deploy Azure IoT Edge modules with Azure CLI

az mysql

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
<code>az mysql flexible-server identity</code>	Migrate an application to use passwordless connections with Azure Database for MySQL
	Configure passwordless database connections for Java apps on Oracle WebLogic Servers
	Use Java and JDBC with Azure Database for MySQL - Flexible Server

az policy

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
<code>az policy assignment identity</code>	Built-in policies for Azure Monitor

az redis

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az redis identity	Managed identity for storage

az spring

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az spring app identity	Migrate a Python application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL
	Migrate an application to use passwordless connections with Azure Service Bus
	Enable system-assigned managed identity for an application in Azure Spring Apps
	Migrate an application to use passwordless connections with Azure Event Hubs
	Migrate a .NET application to use passwordless connections with Azure SQL Database
	Migrate a Node.js application to use passwordless connections with Azure SQL Database
	Migrate a Java application to use passwordless connections with Azure SQL Database
	Migrate an application to use passwordless connections with Azure Database for MySQL

az synapse

[+] Expand table

Reference subgroup	Azure CLI article showing reference use
az synapse workspace managed-identity	Move an Azure Synapse Analytics workspace from one region to another

az vm

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az vm identity	Sign in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	Use an Azure managed identity to authenticate to an Azure container registry
	Use the Linux diagnostic extension 4.0 to monitor metrics and logs
	Connect with Managed Identity to Azure Database for PostgreSQL
	Authenticate Azure-hosted apps to Azure resources with the Azure SDK for .NET
	Migrate a Python application to use passwordless connections with Azure SQL Database
	Send Prometheus metrics from virtual machines, scale sets, or Kubernetes clusters to an Azure Monitor workspace
	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for Python
	Migrate an application to use passwordless connections with Azure Blob Storage
	Migrate an application to use passwordless connections with Azure Database for PostgreSQL

az vmss

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az vmss identity	Sign in to a Linux virtual machine in Azure by using Microsoft Entra ID and OpenSSH
	Fail to pull images from Azure Container Registry to Azure Kubernetes Service cluster
	Use the Linux diagnostic extension 4.0 to monitor metrics and logs
	Send Prometheus metrics from virtual machines, scale sets, or Kubernetes clusters to an Azure Monitor workspace
	Send Prometheus data to Azure Monitor by using managed identity authentication
	Create a chaos experiment that uses an agent-based fault with the Azure CLI
	Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)

az webapp

[+] [Expand table](#)

Reference subgroup	Azure CLI article showing reference use
az webapp identity	How to use managed identities for App Service and Azure Functions
	Tutorial: Connect to SQL Database from .NET App Service without secrets using a managed identity
	Configure a custom container for Azure App Service
	Connect to and query Azure SQL Database using Python and the pyodbc driver
	Migrate custom software to Azure App Service using a custom container
	Tutorial: Connect to Azure databases from App Service without secrets using a managed identity
	Authenticate Azure-hosted apps to Azure resources with the Azure SDK for .NET
	Migrate a Python application to use passwordless connections with Azure

Reference subgroup	Azure CLI article showing reference use
	SQL Database
	Authenticating Azure-hosted apps to Azure resources with the Azure SDK for Python
	Migrate an application to use passwordless connections with Azure Blob Storage

How to create resources at scale using the Azure CLI

Article • 07/30/2024

As an Azure resource manager, you frequently have to create multiple Azure resources when configuring new environments. You might also have an Azure resource approval process that works best when Azure resources are created automatically from a script.

In this article you will learn the following:

- Create multiple Azure resources from parameters received from a delimited CSV file.
- Use IF..THEN statements to create dependent Azure resources.
- Log script progress to a local TXT file.

This sample script has been tested in [Azure Cloud Shell](#) in both Bash and PowerShell environments, and [PowerShell 7](#). Find the CSV and full script in [Azure-samples/azure-cli-samples](#).

Prepare your environment

Follow these steps to prepare your environment to run the example script:

- Open either the Bash or PowerShell environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
- Download and save to a local directory the following CSV file. Replace `myExistingResourceGroupName` in line three with an actual resource group name.

CSV

```
resourceNo,location,createRG,existingRgName,createVnet,vnetAddressPrefix  
,subnetAddressPrefixes,vmImage,publicIpSku,AdminUser  
1,eastus,TRUE,,TRUE,10.0.0.0/16,10.0.0.0/24,Ubuntu2204,standard,  
2,eastus2,TRUE,,FALSE,,,Debian11,standard,alex-smith  
3,southcentralus,FALSE,myExistingResourceGroupName,TRUE,,,Ubuntu2204,s  
tandard,jan-smith  
[empty line for Bash]
```

 Note

To be a proper Unix text file and be read by Bash, the CSV file needs a newline character at the end of the last data line. This results in a blank line at the end of the file. Your blank line does not need to say [empty line] as this text is only provided to show you that an empty line exists. PowerShell environments do not have this newline character requirement.

- Upload your modified CSV file to your Azure Cloud Shell blog storage account. The easiest way to do this is to use the **Manage files** drop down on the Azure Cloud Shell main menu. For more information on Cloud Shell storage, see [Persist files in Azure Cloud Shell](#).

Script overview

This article breaks a single large script into four sections so each step can be explained.

- Variable setup
- Data validation
- Loop validation
- Azure resource creation

There are also two scripts provided: one for Bash and the second for PowerShell. *Both scripts use the same Azure CLI commands.* It is the environment, or terminal profile, that is different. For example, Bash uses `do...done` and `if...then...fi`. In a PowerShell environment, you use the equivalent `foreach` and `if (something is true)...{do this}`. In Azure Cloud Shell you can switch between environments by using the **Switch to PowerShell** or **Switch to Bash** button in the Azure Cloud Shell main menu.

If you prefer, go directly to the CSV and script files used by this article in [Azure-samples/azure-cli-samples](#).

Set variables

Get started by creating the variables needed for the script. The following three variables need actual values for your environment:

- **subscriptionID**: This is your Azure subscription ID.
- **csvFileLocation**: This is the location and file name of your CSV input file.
- **logFileLocation**: This is the location and file name *the script will use to create a log file*. You do not need to create or upload this file.

Variables with a `msdocs-` prefix can be replaced with the prefix of your choice. All empty (`""`) variables use values from the CSV input file. These empty variables are placeholders needed by the script.

```
Bash

Azure CLI

# Variable block

# Replace these three variable values with actual values
subscriptionID=00000000-0000-0000-0000-00000000
csvFileLocation="myFilePath\myFileName.csv"
logFileLocation="myFilePath\myLogName.txt"

# Variable values that contain a prefix can be replaced with the prefix
# of your choice.
# These prefixes have a random ID appended to them in the script.
# Variable values without a prefix will be overwritten by the contents
# of your CSV file.
location=""
createRG=""
newRgName="msdocs-rg-"
existingRgName=""

createVnet=""
vnetName="msdocs-vnet-"
subnetName="msdocs-subnet-"
vnetAddressPrefix=""
subnetAddressPrefixes=""

vmName="msdocs-vm-"
vmImage=""
publicIpSku=""
adminUser=""
adminPassword="msdocs-PW-@"

# Set your Azure subscription
az account set --subscription $subscriptionID
```

Validate CSV file values

Before you start to test the create script, make sure your CSV file is formatted correctly and variables will be assigned correct values. This script uses an IF..THEN statement so you can look at one scenario/CSV line at a time.

```
Bash
```

Azure CLI

```
# Verify CSV columns are being read correctly

# Take a look at the CSV contents
cat $csvFileLocation

# Validate select CSV row values
while IFS=, read -r resourceNo location createRG existingRgName
createVnet vnetAddressPrefix subnetAddressPrefixes vmImage publicIpSku
adminUser
do
    # Generate a random ID
    let "randomIdentifier=$RANDOM*$RANDOM"

    # Return the values for the first data row
    # Change the $resourceNo to check different scenarios in your CSV
    if [ "$resourceNo" = "1" ]; then
        echo "resourceNo = $resourceNo"
        echo "location = $location"
        echo "randomIdentifier = $randomIdentifier"
        echo ""

        echo "RESOURCE GROUP INFORMATION:"
        echo "createRG = $createRG"
        if [ "$createRG" = "TRUE" ]; then
            echo "newRgName = $newRgName$randomIdentifier"
        else
            echo "existingRgName = $existingRgName"
        fi
        echo ""

        echo "VNET INFORMATION:"
        echo "createVnet = $createVnet"
        if [ "$createVnet" = "TRUE" ]; then
            echo "vnetName = $vnetName$randomIdentifier"
            echo "subnetName = $subnetName$randomIdentifier"
            echo "vnetAddressPrefix = $vnetAddressPrefix"
            echo "subnetAddressPrefixes = $subnetAddressPrefixes"
        fi
        echo ""

        echo "VM INFORMATION:"
        echo "vmName = $vmName$randomIdentifier"
        echo "vmImage = $vmImage"
        echo "vmSku = $publicIpSku"
        if [ `expr length "$adminUser"` == "1" ]; then
            echo "SSH keys will be generated."
        else
            echo "vmAdminUser = $adminUser"
            echo "vmAdminPassword = $adminPassword$randomIdentifier"
        fi
    fi
fi
```

```
# skip the header line
done < <(tail -n +2 $csvFileLocation)
```

Using the CSV provided in this article, the validation output is as follows: (The `00000001` random ID will be different for each test.)

Output

```
resourceNo = 1
location = eastus

RESOURCE GROUP INFORMATION:
createRG = TRUE
newRGName = msdocs-rg-00000001

VNET INFORMATION:
createVnet = TRUE
vnetName = msdocs-vnet-00000001
subnetName = msdocs-subnet-00000001
vnetAddressPrefix = 10.0.0.0/16
subnetAddressPrefix = 10.0.0.0/24

VM INFORMATION:
vmName = msdocs-vm-00000001
vmImage = Ubuntu2204
vmSku = standard
SSH keys will be created
```

Validate script logic

If you are confident in your scripting abilities, you can skip this step. However, because this script is designed to create Azure resources at scale, looping through the script with `echo` or `write-host` statements can save you time and unexpected billable Azure resources.

Bash

There are several ways to iterate through a CSV file using Bash. This example uses `IFS` with a `while` loop.

Azure CLI

```
# Validate script logic

# Create the log file
echo "SCRIPT LOGIC VALIDATION.">$logFileLocation
```

```

# Loop through each row in the CSV file
while IFS=, read -r resourceNo location createRG existingRgName
createVnet vnetAddressPrefix subnetAddressPrefixes vmImage publicIpSku
adminUser
do
    # Generate a random ID
    let "randomIdentifier=$RANDOM*$RANDOM"

    # Log resource number and random ID
    echo "resourceNo = $resourceNo">>>$logFileLocation
    echo "randomIdentifier = $randomIdentifier">>>$logFileLocation

    # Check if a new resource group should be created
    if [ "$createRG" == "TRUE" ]; then
        echo "Will create RG $newRgName$randomIdentifier.">>$logFileLocation
        existingRgName=$newRgName$randomIdentifier
    fi

    # Check if a new virtual network should be created, then create the VM
    if [ "$createVnet" == "TRUE" ]; then
        echo "Will create VNet $vnetName$randomIdentifier in RG
$existingRgName.">>$logFileLocation
        echo "Will create VM $vmName$randomIdentifier in Vnet
$vnetName$randomIdentifier in RG $existingRgName.">>$logFileLocation
    else
        echo "Will create VM $vmName$randomIdentifier in RG
$existingRgName.">>$logFileLocation
    fi
    # Skip the header line.
done < <(tail -n +2 $csvFileLocation)

# Clear the console and display the log file
Clear
cat $logFileLocation

```

Using the CSV provided in this article, the validation output is as follows: (The `00000001`, `2`, `3` random IDs will be different for each test, but each resource under each `resourceNo` should share the same random ID.)

Output

```

resourceNo = 1
createRG = TRUE
createVnet = TRUE
Will create RG msdocs-rg-00000001
Will create VNet msdocs-vnet-00000001 in RG msdocs-rg-00000001
Will create VM msdocs-vm-00000001 within Vnet msdocs-vnet-00000001 in RG
msdocs-rg-00000001

resourceNo = 2

```

```
createRG = TRUE
createVnet = FALSE
Will create RG msdocs-rg-00000002
Will create VM msdocs-vm-00000002 without Vnet in RG msdocs-rg-00000002

resourceNo = 3
createRG = FALSE
createVnet = FALSE
Will create VM msdocs-vm-00000003 without Vnet in RG
<myExistingResourceGroup>
```

Create Azure resources

You have now created your variable block, validated your CSV values, and completed a test run with `echo` or `write-host`. Execute the fourth and final portion of the script to create Azure resources as defined in your CSV input file.

Bash

Azure CLI

```
# Create Azure resources

# Create the log file
echo "CREATE AZURE RESOURCES.">$logFileLocation

# Loop through each CSV row
while IFS=, read -r resourceNo location createRG existingRgName
    createVnet vnetAddressPrefix subnetAddressPrefixes vmImage publicIpSku
    adminUser
do
    # Generate a random ID
    let "randomIdentifier=$RANDOM*$RANDOM"

    # Log resource number, random ID and display start time
    echo "resourceNo = $resourceNo">>$logFileLocation
    echo "randomIdentifier = $randomIdentifier">>$logFileLocation
    echo "Starting creation of resourceNo $resourceNo at $(date +"%Y-%m-%d %T")."

    # Check if a new resource group should be created
    if [ "$createRG" == "TRUE" ]; then
        echo "Creating RG $newRgName$randomIdentifier at $(date +"%Y-%m-%d %T").">>$logFileLocation
        az group create --location $location --name
        $newRgName$randomIdentifier >>$logFileLocation
        existingRgName=$newRgName$randomIdentifier
        echo "  RG $newRgName$randomIdentifier creation complete"
    fi
```

```

# Check if a new virtual network should be created, then create the VM
if [ "$createVnet" == "TRUE" ]; then
    echo "Creating VNet $vnetName$randomIdentifier in RG $existingRgName
at $(date +"%Y-%m-%d %T").">>$logFileLocation
    az network vnet create \
        --name $vnetName$randomIdentifier \
        --resource-group $existingRgName \
        --address-prefix $vnetAddressPrefix \
        --subnet-name $subnetName$randomIdentifier \
        --subnet-prefixes $subnetAddressPrefixes >>$logFileLocation
    echo "  VNet $vnetName$randomIdentifier creation complete"

    echo "Creating VM $vmName$randomIdentifier in Vnet
$vnetName$randomIdentifier in RG $existingRgName at $(date +"%Y-%m-%d
%T").">>$logFileLocation
    az vm create \
        --resource-group $existingRgName \
        --name $vmName$randomIdentifier \
        --image $vmImage \
        --vnet-name $vnetName$randomIdentifier \
        --subnet $subnetName$randomIdentifier \
        --public-ip-sku $publicIpSku \
        --generate-ssh-keys >>$logFileLocation
    echo "  VM $vmName$randomIdentifier creation complete"
else
    echo "Creating VM $vmName$randomIdentifier in RG $existingRgName at
$(date +"%Y-%m-%d %T").">>$logFileLocation
    az vm create \
        --resource-group $existingRgName \
        --name $vmName$randomIdentifier \
        --image $vmImage \
        --public-ip-sku $publicIpSku \
        --admin-username $adminUser \
        --admin-password $adminPassword$randomIdentifier
>>$logFileLocation
    echo "  VM $vmName$randomIdentifier creation complete"
fi
# skip the header line
done < <(tail -n +2 $csvFileLocation)

# Clear the console (optional) and display the log file
# clear
cat $logFileLocation

```

In your console output, are you missing the last row in your CSV file? This can be caused by a missing line continuation character after the last line. Add a blank line at the end of your CSV file to fix the issue.

Console output *before* log file read:

Output

```
Starting creation of resourceNo 1 at YYYY-MM-DD HH:MM:SS.  
RG msdocs-rg-00000001 creation complete  
VNet msdocs-vnet-00000001 creation complete  
VM msdocs-vm-00000001 creation complete  
  
Starting creation of resourceNo 2 at YYYY-MM-DD HH:MM:SS.  
RG msdocs-rg-00000002 creation complete  
VM msdocs-vm-00000002 creation complete  
  
Starting creation of resourceNo 3 at YYYY-MM-DD HH:MM:SS.  
VM msdocs-vm-00000003 creation complete
```

Your log file contents should look similar to this:

```
Output  
  
Starting creation of resourceNo 1 at YYYY-MM-DD HH:MM:SS.  
Creating RG msdocs-rg-00000001 at YYYY-MM-DD HH:MM:SS.  
{  
Resource group create output  
}  
Creating VNet msdocs-vnet-00000001 in RG msdocs-rg-00000001 at YYYY-MM-DD  
HH:MM:SS.  
{  
VNet create output  
}  
Creating VM msdocs-vm-00000001 in RG msdocs-rg-00000001 at YYYY-MM-DD  
HH:MM:SS.  
{  
VM create output  
}  
  
Starting creation of resourceNo 2 at YYYY-MM-DD HH:MM:SS.  
Creating RG msdocs-rg-00000002 at YYYY-MM-DD HH:MM:SS.  
{  
Resource group create output  
}  
Creating VM msdocs-vm-00000002 in RG msdocs-rg-00000002 at YYYY-MM-DD  
HH:MM:SS.  
{  
VM create output  
}  
  
Starting creation of resourceNo 3 at YYYY-MM-DD HH:MM:SS.  
Creating msdocs-vm-00000003 creation complete  
{  
VM create output  
}
```

Troubleshooting

In Bash, the "Create Azure resources" step stops after step 1

In Ubuntu 22.04.3 LTS and Debian version 12 (bookworm) the [Validate script logic](#) works as expected returning results for all three resources. However, the [Create Azure resources](#) stops after the first resource. A possible reason for this issue is that the creating the VNet in step #1 takes a few seconds. Both Ubuntu and Debian proceed to the second resource without waiting for the completion of the VNet. You can read more about this in [wait doesn't wait for the processes in the while loop to finish ↗](#) or [Waiting for any process to finish in bash script ↗](#).

Bash script ignores IF statement

Bash is case sensitive. The word `true` does not equal `TRUE`. Also `greater than` is `-gt`, not `>`, and `equals` is `==`, not `=`. Make sure you do not have a typographical error, or leading/trailing spaces in your CSV column values.

Variable values are not changing with each loop

This is often caused by extra spaces in the CSV file. A line in a CSV file will look something like this: `column1,column2,column3` or `column1,,column3`, but by habit it is easy to create a test file that contains a space after each comma like `column1, column2, column3`. When you have a leading or trailing space in your CSV, the column value is actually `<space>columnValue`. The script logic `if ["$columnName" = "columnValue"]` returns "false". Remove all leading and trailing spaces in your CSV rows to fix the issue.

Invalid CIDR notation

You receive an `InvalidCIDRNotation` error when you pass an incorrect address prefix to `az network vnet create`. This can be challenging when visually, the address prefix looks correct when returned in an `echo` statement. To troubleshoot the actual value being read from the CSV, try this script:

Azure CLI

```
while IFS=, read -r resourceNo location createRG existingRgName createVnet
vnetAddressPrefix subnetAddressPrefixes vmImage publicIpSku adminUser
do
    echo "resourceNo = $resourceNo"

    if [ "$createVnet" == "TRUE" ]; then
        startTest="abc"
```

```
    endTest="xyz"
    echo $startTest$vnetAddressPrefix$endTest
  fi
done < <(tail -n +2 $setupFileLocation)
```

If your results look like `xzy10.0.0.0` and not the expected `abc10.0.0.0/24xyz`, there might be a hidden character or extra comma lurking in your CSV file. Add a test column with the same prefix value, rearrange your CSV columns, and copy/paste your CSV contents in/out of a simple Notepad editor. In writing this article, the rearrangement of the CSV columns finally fixed the error.

Arguments are expected or required

You receive this error when you have not supplied a required parameter or there is a typographical error that causes the Azure CLI to incorrectly parse the reference command. When working with a script, you also receive this error when one of more of the following is true:

- There is a missing or incorrect line continuation character.
- There are trailing spaces on the right side of a line continuation character.
- Your variable name contains a special character, such as a dash (-).

InvalidTemplateDeployment

When you try to create an Azure resource in a location that does not offer that resource you receive an error similar to the following: "Following SKUs have failed for Capacity Restrictions: Standard_DS1_v2' is currently not available in location 'westus'."

Here's the full error example:

Error

```
{"error":{"code":"InvalidTemplateDeployment","message":"The template deployment 'vm_deploy_<32 character ID>' is not valid according to the validation procedure. The tracking id is '<36 character ID>'. See inner errors for details.","details":[{"code":"SkuNotAvailable","message":"The requested VM size for resource 'Following SKUs have failed for Capacity Restrictions: Standard_DS1_v2' is currently not available in location '<your specified location>'. Please try another size or deploy to a different location or different zone. See https://aka.ms/azureskunotavailable for details."}]}}
```

To correct the error, either change the location or select a different parameter value that is offered for your desired location.

See also

- [Delete Azure resources at scale](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

How to delete resources at scale using the Azure CLI

Article • 05/27/2024

As an Azure resource manager, you frequently have to delete multiple Azure resources when tearing down an old environment. Some CLI devTest environments also need a periodic clean up so charges are not incurred for temporary Azure resources that have lingered longer.

In this Azure CLI sample you will learn the following:

- ✓ Delete multiple Azure resources from a script
- ✓ Log script progress to a local TXT file

This sample script has been tested in [Azure Cloud Shell](#) in both Bash and PowerShell environments. This script has also been tested successfully in [PowerShell 7](#) and [Windows Subsystem for Linux \(WSL\)](#) with Ubuntu 22.04.3 LTS using [Windows Terminal](#).

Delete Azure resources filtering by name

Use this script to list and delete resource groups that start with a given word.

```
Azure CLI

# Set your subscription
subscriptionID=00000000-0000-0000-0000-00000000
az account set --subscription $subscriptionID

# Set your log file location
logFileLocation="myLogName.txt"

# Get the name of all resource groups that start with 'msdocs'
az group list --query "[?starts_with(name, 'msdocs') == \`true\`].name" -o
table

# Delete resource groups without a confirmation prompt (--yes)
# Do not wait for the operation to finish (--no-wait)
echo "Deleting resource groups">>$logFileLocation
for rgList in $(az group list --query "[?starts_with(name, 'msdocs') ==
\`true\`].name" -o tsv);
do
    echo "deleting resource group $rgList">>>$logFileLocation
    az group delete --name $rgList --yes --no-wait
done

# read your log file with Linux "cat" command
```

```
clear  
cat $logFileLocation
```

Delete Azure resources filtering by creation date

Use this script to list and delete storage accounts that were created within a date range.

Azure CLI

```
# Set your log file location  
logFileLocation="myLogName.txt"  
  
# Set your resource group variable  
rgName=<msdocs-rg-0000000>  
  
# Get a list of Azure storage accounts that were created in the last 30  
days. Return the results as a table.  
saDate=$(date +%F -d "-30days")  
az storage account list --resource-group $rgName \  
--query "[?creationTime >='$saDate'].{saName:name,  
createdTimeStamp:creationTime}" \  
--output table  
  
# Delete storage accounts without a confirmation prompt (--yes).  
# Do not wait for the operation to finish (--no-wait)  
echo "Deleting storage accounts">>$logFileLocation  
for saList in $(az storage account list --resource-group $rgName \  
--query "[?creationTime >='$saDate'].{saName:name,  
createdTimeStamp:creationTime}" \  
--output tsv);  
do  
    echo "deleting storage account $saList">>>$logFileLocation  
    az storage account delete --ids $saList --yes --no-wait  
done  
  
# read your log file with Linux "cat" command  
clear  
cat $logFileLocation
```

Delete all azure resources of a type

Delete all Virtual Machines in a resource group

Azure CLI

```
# Set your resource group variable
rgName=<msdocs-rg-0000000>

az group delete -n $rgName --force-deletion-types
Microsoft.Compute/virtualMachines
```

See also

- [Create Azure resources at scale](#)

How to query Azure CLI command output using a JMESPath query

Article • 06/27/2024

The Azure CLI uses the `--query` parameter to execute a [JMESPath query](#) on the results of commands. JMESPath is a query language for JSON, giving you the ability to select and modify data from CLI output.

All commands in Azure CLI support the `--query` parameter. This article covers how to use the features of JMESPath and gives examples of queries. Learn about JMESPath concepts that are useful for querying under the concepts tab. See examples of JMESPath queries under the examples tab.

Concepts

Azure CLI uses queries to select and modify the output of Azure CLI commands. Queries are executed client-side on the Azure CLI command's returned JSON object before any display formatting.

The escape characters needed in queries differ for different environments. It's recommended to run queries in Azure Cloud Shell or cmd because these shells require fewer escape characters. To ensure the query examples are syntactically correct, select the tab for the shell you're using.

Dictionary and list CLI results

CLI command results are first treated as JSON for queries, even when the output format is something other than JSON. CLI results are either a JSON array or dictionary. Arrays are sequences of objects that can be indexed, and dictionaries are unordered objects accessed with keys.

This is an example of an array:

JSON

```
[  
  1,  
  2,  
  3  
]
```

This is an example of a dictionary:

JSON

```
{  
  "isRunning": false,  
  "time": "12:00",  
  "number": 1  
}
```

Commands that *could* return more than one object return an array, and commands that *always* return *only* a single object return a dictionary.

Get properties in a dictionary

Working with dictionary results, you can access properties from the top level with just the key. The `.` (subexpression) character is used to access properties of nested dictionaries. Before introducing queries, take a look at the unmodified output of the [az vm show](#) command:

Bash

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM
```

The command outputs a dictionary. Some content has been omitted.

JSON

```
{  
  "additionalCapabilities": null,  
  "availabilitySet": null,  
  "diagnosticsProfile": {  
    "bootDiagnostics": {  
      "enabled": true,  
      "storageUri": "https://xxxxxx.blob.core.windows.net/"  
    }  
  },  
  ...  
  "osProfile": {  
    "adminPassword": null,  
    "adminUsername": "azureuser",  
    "allowExtensionOperations": true,  
    "computerName": "TestVM",  
    "customData": null,  
    "imageReference": {  
      "id": "https://xxxxxxxxxxxxxx.vhdsblo  
      "label": "Windows Server 2016 Data Center",  
      "publisher": "Microsoft",  
      "offer": "WindowsServer",  
      "sku": "Datacenter",  
      "version": "latest"  
    },  
    "osDisk": {  
      "caching": "None",  
      "createOption": "FromImage",  
      "managedDisk": {  
        "id": "https://xxxxxxxxxxxxxx.vhdsblo  
        "storageAccountType": "Standard_LRS",  
        "useExisting": false  
      },  
      "name": "osdisk",  
      "osType": "Windows",  
      "vhd": {  
        "uri": "https://xxxxxxxxxxxxxx.vhdsblo  
      }  
    },  
    "secrets": [{}],  
    "storageAccounts": [{}],  
    "ultraVSS": false  
  }  
}
```

```

    "linuxConfiguration": {
        "disablePasswordAuthentication": true,
        "provisionVmAgent": true,
        "ssh": [
            {
                "publicKeys": [
                    {
                        "keyData": "ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQDMobZNJTqgjWn/IB5xli1vE4Y+BMYpqkDnGRUcA0g9
BYPgrGSQuCES37v2e3JmpfDPHFsaR+CPK1Vr2GoVJMMHeRcMJhj50ZWq0hAnkJBh1ZVWy8S
7dwdGAqPyPmWM2iJDCVMVrLITAJCno4704Ees7RCH6ku7kU86b1NOanvrNwqTHr14wtnLhgZ
0gQ5GV1oLWvMEVg1YFMIgPRkTsSQKWC51LqQ45aU/4NMJoUxGyJTL9i8YxMavaB1Z2npfTQ
DQo9+womZ7SXzHaIWC858gWNl9e5UFyHDnTEDc14hKkf1CqnGJVcCJkmSfmrrHk/CkmF0ZT3
whTH01DhJTtV stramer@contoso",
                        "path": "/home/azureuser/.ssh/authorized_keys"
                    }
                ]
            },
            "secrets": [],
            "windowsConfiguration": null
        ],
        ....
    }
}

```

The following command gets the SSH public keys authorized to connect to the VM by adding a query:

Bash

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM --query
"osProfile.linuxConfiguration.ssh.publicKeys"
```

JSON

```
[
{
    "keyData": "ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQDMobZNJTqgjWn/IB5xli1vE4Y+BMYpqkDnGRUcA0g9
BYPgrGSQuCES37v2e3JmpfDPHFsaR+CPK1Vr2GoVJMMHeRcMJhj50ZWq0hAnkJBh1ZVWy8S
7dwdGAqPyPmWM2iJDCVMVrLITAJCno4704Ees7RCH6ku7kU86b1NOanvrNwqTHr14wtnLhgZ
0gQ5GV1oLWvMEVg1YFMIgPRkTsSQKWC51LqQ45aU/4NMJoUxGyJTL9i8YxMavaB1Z2npfTQ
DQo9+womZ7SXzHaIWC858gWNl9e5UFyHDnTEDc14hKkf1CqnGJVcCJkmSfmrrHk/CkmF0ZT3
whTH01DhJTtV stramer@contoso",
    "path": "/home/azureuser/.ssh/authorized_keys"
}]
```

Query strings are case sensitive. For example, changing 'osProfile' to 'OsProfile' in the previous query doesn't return the correct results.

Get multiple values

To get more than one property, put expressions separated by commas in square brackets [] (a **multiselect list**). The following command gets the VM name, admin user, and SSH key all at once:

Bash

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM --query "[name,  
osProfile.adminUsername,  
osProfile.linuxConfiguration.ssh.publicKeys[0].keyData]"
```

JSON

```
[  
  "TestVM",  
  "azureuser",  
  "ssh-rsa  
AAAAAB3NzaC1yc2EAAAQABAAQDMobZNJTqgjWn/IB5xli1vE4Y+BMYpqkDnGRUcA0g9  
BYPgrGSQuCES37v2e3JmpfDPHFsaR+CPK1Vr2GoVJMMHeRcMJhj50ZWq0hAnkJBh1ZVWy8S  
7dwdGAqPyPmWM2iJDCVMvrLITAJCno4704Ees7RCH6ku7kU86b1NOanvrNwqTHr14wtnLhgZ  
0gQ5GV1oLWvMEvg1YFMIgPRkTsSQKWC51LqQ45aU/4NMJoUxGyJTL9i8YxMavaB1Z2npfTQ  
DQo9+womZ7SXzHaIWC858gWN19e5UFyHDnTEDc14hKkf1CqnGJVcCJkmSfmrrHk/CkmF0ZT3  
whTH01DhJTtV stramer@contoso"  
]
```

These values are listed in the result array in the order they were given in the query. Since the result is an array, there are no keys associated with the results. To get a dictionary instead of an array, see the next section.

Rename properties in a query

To get a dictionary instead of an array when querying for multiple values, use the { } (multiselect hash) operator. The format for a multiselect hash is {
 displayName: JMESPathExpression, ... }. `displayName` is the string shown in output, and `JMESPathExpression` is the JMESPath expression to evaluate. Modify the example from the last section by changing the multiselect list to a hash:

① Note

If you choose to use a space in a new column name, like `VM name` instead of `VMName`, the quoting rules change in both Bash and PowerShell. See [Pass spaces in Azure CLI parameters](#) for examples.

Bash

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM --query "
{VMName:name, admin:osProfile.adminUsername,
sshKey:osProfile.linuxConfiguration.ssh.publicKeys[0].keyData}"
```

JSON

```
{
  "VMName": "TestVM",
  "admin": "azureuser",
  "ssh-key": "ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQDMobZNJTqgjWn/IB5xli1vE4Y+BMYpqkDnGRUcA0g9
BYPgrGSQuCES37v2e3JmpfDPHFsaR+CPK1Vr2GoVJMMHeRcMJhj50ZWq0hAnkJBh1ZVWy8S
7dwdGAqPyPmWM2iJDCVMVrLITAJCno4704Ees7RCH6ku7kU86b1NOanvrNwqTHr14wtnLhgZ
0gQ5GV1oLWvMEVg1YFMIgPRkTsSQKWC51LqQ45aU/4NMJoUxGyJTL9i8YxMavaB1Z2npfTQ
DQo9+womZ7SXzHaIWc858gWN19e5UFyHDnTEDc14hKkf1CqnGJVcCJkmSfmrrHk/CkmF0ZT3
whTHO1DhJTtV stramer@contoso"
}
```

Get properties in an array

An array has no properties of its own, but it can be indexed. This feature is shown in the last example with the expression `publicKeys[0]`, which gets the first element of the `publicKeys` array. There's no guarantee CLI output is ordered, so avoid using indexing unless you're sure of the order or don't care which element you get. To access the properties of elements in an array, you do one of two operations: *flattening* or *filtering*. This section covers how to flatten an array.

Flattening an array is done with the `[]` JMESPath operator. All expressions after the `[]` operator are applied to each element in the current array. If `[]` appears at the start of the query, it flattens the CLI command result. The results of `az vm list` can

be inspected with this feature. The following query gets the name, OS, and administrator name for each VM in a resource group:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[].{Name:name, OS:storageProfile.osDisk.osType, admin:osProfile.adminUsername}"
```

JSON

```
[  
  {  
    "Name": "Test-2",  
    "OS": "Linux",  
    "admin": "sttramer"  
  },  
  {  
    "Name": "TestVM",  
    "OS": "Linux",  
    "admin": "azureuser"  
  },  
  {  
    "Name": "WinTest",  
    "OS": "Windows",  
    "admin": "winadmin"  
  }  
]
```

Any array can be flattened, not just the top-level result returned by the command.

In the last section, the expression

`osProfile.linuxConfiguration.ssh.publicKeys[0].keyData` was used to get the SSH public key for sign-in. To get *every* SSH public key, the expression could instead be written as `osProfile.linuxConfiguration.ssh.publicKeys[].keyData`. This query expression flattens the `osProfile.linuxConfiguration.ssh.publicKeys` array, and then runs the `keyData` expression on each element:

Bash

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM --query "  
  {VMName:name, admin:osProfile.adminUsername,}
```

```
sshKeys:osProfile.linuxConfiguration.ssh.publicKeys[].keyData }"
```

JSON

```
{  
    "VMName": "TestVM",  
    "admin": "azureuser",  
    "sshKeys": [  
        "ssh-rsa  
AAAAB3NzaC1yc2EAAAQABAAQDMobZNJTqgjWn/IB5xliIvE4Y+BMYpqkDnGRUcA0g9  
BYPgrGSQquCES37v2e3JmpfDPHFsaR+CPKlVr2GoVJMMHeRcMJhj50ZWq0hAnkJBh1ZVWy8S  
7dwdGAqPyPmWM2iJDCVMVrLITAJCno4704Ees7RCH6ku7kU86b1NOanvrNwqTHr14wtnLhgZ  
0gQ5GV1oLWvMEVg1YFMIgPRkTsSQKWCG51LqQ45aU/4NMJoUxGyJTL9i8YxMavaB1Z2npfTQ  
DQo9+womZ7SXzHaIWC858gwN19e5UFyHDnTEDc14hKkf1CqnGJVcCJkmSfmrrHk/CkmF0ZT3  
whTH01DhJTtV stramer@contoso\n"  
    ]  
}
```

Filter arrays with boolean expressions

The other operation used to get data from an array is *filtering*. Filtering is done with the `[?...]` JMESPath operator. This operator takes a predicate as its contents. A predicate is any statement (including Boolean properties) that can be evaluated to either `true` or `false`. Expressions where the predicate evaluates to `true` are included in the output.

The first query demonstrates how to list the names of all Azure subscriptions connected to your account whose `isDefault` property is true. The second and third queries show two different ways to list all subscriptions whose `isDefault` property is false.

Bash

Azure CLI

```
# Boolean values are assumed to be true, so you can directly  
evaluate the isDefault property to return the default subscription.  
az account list --query "[?isDefault].name"  
  
# To check if a Boolean property is false, you can use the  
comparison operator == or the logical operator !.  
az account list --query '[?!isDefault].name'  
az account list --query "[?isDefault == \`false\`].name"
```

JMESPath offers the standard comparison and logical operators. These include `<`, `<=`, `>`, `>=`, `==`, and `!=`. JMESPath also supports logical and (`&&`), or (`||`), and not (`!`). Expressions can be grouped within parenthesis, allowing for more complex predicate expressions. For the full details on predicates and logical operations, see the [JMESPath specification](#).

In the last section, you flattened an array to get the complete list of all VMs in a resource group. With the use of filters, this output can be restricted to only Linux VMs:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[?storageProfile.osDisk.osType=='Linux'].{Name:name, admin:osProfile.adminUsername}" --output table
```

JSON

Name	Admin
Test-2	sttramer
TestVM	azureuser

You can also filter numerical values such as the OS disk size. The following example demonstrates how to filter the list of VMs to display ones with a disk size larger than or equal to 50 GB.

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[?storageProfile.osDisk.diskSizeGb >=\`50\`].{Name:name, admin:osProfile.adminUsername, DiskSize:storageProfile.osDisk.diskSizeGb }" --output table
```

JSON

Name	Admin	DiskSize
------	-------	----------

```
WinTest winadmin 127
```

For large arrays, it may be faster to apply the filter before selecting data.

ⓘ Important

In JMESPath, strings are always surrounded by single quotes (') or escape characters (`). If you use double quotes as part of a string in a filter predicate, you'll get empty output.

JMESPath functions

JMESPath also has built-in functions that allow for more complex queries and for modifying query output. This section focuses on using JMESPath functions to create queries while the [Manipulating output with functions](#) section demonstrates how to use functions to modify the output.

Expressions are evaluated before calling the function, so arguments themselves can be JMESPath expressions. The following examples demonstrate this concept by using `contains(string, substring)`, which checks to see if a string contains a substring. This command finds all VMs using SSD storage for their OS disk:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[?
contains(storageProfile.osDisk.managedDisk.storageAccountType, 'SSD')
].{Name:name,
Storage:storageProfile.osDisk.managedDisk.storageAccountType}"
```

JSON

```
[{
  "Name": "TestVM",
  "Storage": "StandardSSD_LRS"
},
{
  "Name": "WinTest",
  "Storage": "StandardSSD_LRS"
```

```
}
```

Pipe expressions

Similar to how `|` is used in the command line, `|` can be used in JMESPath queries to apply expressions to intermediate query results. We can also use `|` to break down complex queries into simpler subexpressions. To shorten the query from the previous section, use `|` to apply the filter after flattening and selecting data.

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[].{Name:name,  
Storage:storageProfile.osDisk.managedDisk.storageAccountType} | [?  
contains(Storage, 'SSD')]"
```

JSON

```
[  
 {  
   "Name": "TestVM",  
   "Storage": "StandardSSD_LRS"  
 },  
 {  
   "Name": "WinTest",  
   "Storage": "StandardSSD_LRS"  
 }]
```

See the [JMESPath specification - Built-in Functions](#) for the full list of functions.

Manipulating output with functions

JMESPath functions also have another purpose, which is to operate on the results of a query. Any function that returns a nonboolean value changes the result of an expression. For example, you can sort data by a property value with `sort_by(array, &sort_expression)`. JMESPath uses a special operator, `&`, for expressions that should be evaluated later as part of a function. The next example shows how to sort a VM list by OS disk size:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "sort_by([].{Name:name, Size:storageProfile.osDisk.diskSizeGb}, &Size)" --output table
```

JSON

Name	Size
-----	-----
Test-2	30
TestVM	32
WinTest	127

See the [JMESPath specification - Built-in Functions](#) for the full list of functions.

Formatting query results

The Azure CLI uses JSON as its default output format, however different output formats may better suit a query depending on its purpose and results. Queries are always run on the `JSON` output first and then formatted.

This section will go over `tsv` and `table` formatting and some use cases for each format. For more information about output formats, see [Output formats for Azure CLI commands](#).

TSV output format

The `tsv` output format returns tab- and newline-separated values without extra formatting, keys, or other symbols. This format is useful when the output is stored in a parameter and used in another command.

One use case for `tsv` formatting is queries that retrieve a value out of a CLI command, such as an Azure resource ID or resource name, and store the value in a local environment variable. By default the results are returned in JSON format, which may be an issue when dealing with JSON strings that are enclosed in `"` characters. The quotes may **not** be interpreted by the shell if the command output is directly assigned to the environment variable. This issue can be seen in the following example that assigns a query result to an environment variable:

Bash

Azure CLI

```
USER=$(az vm show --resource-group QueryDemo --name TestVM --query  
"osProfile.adminUsername")  
echo $USER
```

JSON

```
"azureuser"
```

Use `tsv` formatting, as demonstrated in the following query, to prevent enclosing return values with type information:

Bash

Azure CLI

```
USER=$(az vm show --resource-group QueryDemo --name TestVM --query  
"osProfile.adminUsername" --output tsv)  
echo $USER
```

JSON

```
azureuser
```

Table output format

The `table` format prints output as an ASCII table, making it easy to read and scan. Not all fields are included in the table so this format is best used as a human-searchable overview of data. Fields that aren't included in the table can still be filtered for as part of a query.

ⓘ Note

Certain keys are filtered out and not printed in the table view. These keys are `id`, `type`, and `etag`. To see these values, you can change the key name in a multiselect hash.

Azure CLI

```
az vm show --resource-group QueryDemo --name TestVM --query "  
{objectID:id}" --output table
```

We can use a previous query to demonstrate this concept. The original query returned a JSON object containing the name, OS, and administrator name for each VM in the resource group:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[].{Name:name,  
OS:storageProfile.osDisk.osType, admin:osProfile.adminUsername}"
```

JSON

```
[  
 {  
   "Name": "Test-2",  
   "OS": "Linux",  
   "admin": "sttramer"  
 },  
 {  
   "Name": "TestVM",  
   "OS": "Linux",  
   "admin": "azureuser"  
 },  
 {  
   "Name": "WinTest",  
   "OS": "Windows",  
   "admin": "winadmin"  
 }]
```

When combined with the `--output table` output format, the column names match up with the `displayKey` value of the multiselect hash making it easier to skim the information:

Bash

Azure CLI

```
az vm list --resource-group QueryDemo --query "[].{Name:name, OS:storageProfile.osDisk.osType, Admin:osProfile.adminUsername}" --output table
```

JSON

Name	OS	Admin
Test-2	Linux	sttramer
TestVM	Linux	azureuser
WinTest	Windows	winadmin

Next steps

To learn more about JMESPath queries, see [JMESPath Tutorial](#).

To learn more about other Azure CLI concepts mentioned in this article see:

- [Output formats for Azure CLI commands](#)
- [How to use Azure CLI effectively](#)
- [Learn to use Bash with Azure CLI](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

How to use shorthand syntax with Azure CLI

Article • 09/23/2024

Many Azure CLI parameters accept JSON values, but because it can be challenging to flatten JSON into a string, the Azure CLI supports shorthand syntax. Shorthand syntax is a simplified representation of a JSON string.

This article provides examples of shorthand syntax, and shows how to pass JSON as a file.

ⓘ Note

You'll know that a parameter accepts shorthand syntax when the parameter description reads similar to `Support shorthand-syntax, JSON-file and YAML-file.`
Try `???` to show more.

Full value shorthand syntax

Shorthand syntax in `Full Value` format is JSON surrounded by *double quotes*. Double quotes cause the JSON value to be passed as a string in both the PowerShell and Bash scripting language. Here's a JSON example:

JSON

```
{  
  "name": "Bill",  
  "age": 20,  
  "paid": true,  
  "emails": [  
    "Bill@microsoft.com",  
    "Bill@outlook.com"  
,  
  "address": {  
    "country": "USA",  
    "company": "Microsoft",  
    "details": {  
      "line1": "15590 NE 31st St",  
      "line2": "Redmond, WA"  
    }  
  }  
}
```

When you pass JSON in a parameter value, the JSON is flattened and wrapped in double quotes.

Bash

```
az some-command --contact "{name:Bill,age:20,paid:true,emails:[Bill@microsoft.com,Bill@outlook.com],address:{country:USA,company:Microsoft,details:{line1:'15590 NE 31st St',line2:'Redmond, WA'}}}"
```

Partial value shorthand syntax

Shorthand syntax for partial value is composed of two parts joined by an equal (=) sign between the index `key` and the `value`. Here's an example: `key=value`.

The value can be a simplified string, full value format, JSON, or a JSON file path. Using the JSON example provided, pass properties for the `--contact` parameter following these examples:

Use `Partial Value` for a single key-value pair.

Bash

```
az some-command --contact name=Bill
```

Use `Partial Value` for two key-value pairs. Note the space that separates the two pairs.

Bash

```
az some-command --contact age=20 paid=true
```

Use `Partial Value` for second element.

Bash

```
az some-command --contact emails[1]="Bill@outlook.com"
```

Use `Partial Value` for *details* property of *address*.

Bash

```
az some-command --contact address.details="{line1:'15590 NE 31st St',line2:'Redmond, WA'}
```

Combine full and partial value shorthand syntax

You can combine full and partial value syntax, but always start with `Full Value` followed by `Partial Value`. If you reverse the order, the final data only contains the `Full Value` without properties defined in `Partial Value`.

Use `Full Value` followed by `Partial Value`:

Bash

```
az some-command --contact "{name:Bill,age:20,paid:true,emails:[Bill@microsoft.com,Bill@outlook.com]}" motto="One man's bug is another man's lesson."
```

You can also patch a new element of list property in `Full Value`. For example you can set the second email address by `Partial Value`:

Bash

```
az some-command --contact "{name:Bill,age:20,paid:true,emails:[Bill@microsoft.com]}" emails[1]="Bill@outlook.com" motto="One man's bug is another man's lesson."
```

Pass JSON in a file

It's also possible to pass a JSON file as a parameter value. This is the recommended approach when working with complex JSON.

Bash

```
az some-command --contact address.details=../address_details.JSON
```

Single Quotes String

Single Quotes String is used to pass a `string` value with special characters: `:`, `,`, `{`, `}`, `[`, `]`, `null`, `??`, and space. These characters often have other meanings when parsing shorthand syntax. Single quotes tell the parser to treat everything as a string.

Pass a string value with a space and other special characters

Here's example JSON to pass as a value in the `--contact` parameter:

```
JSON

{
  "name": "Bill RP",
  "age": 20,
  "paid": true,
  "data": "{a: [1, 2]}"
}
```

Use single quotes in `Full Value` format:

```
Bash

az some-command --contact "{name:'Bill RP',age:20,paid:true,data:'{a: [1, 2]}'}"
```

Use single quotes in `Partial Value` format:

```
Bash

az some-command --contact name="'Bill RP'" data="'{a: [1, 2]}'"
```

In the next example, it's also possible to remove single quotes for the `name` key. The parser doesn't distinguish between a `Full Value` expression, `null` value, or the `??` flag.

Use `Partial Value` format:

```
Bash

az some-command --contact name="Bill RP"
```

Work with apostrophes

The apostrophe character (`'`) needs special escape (`/'`) in *Single Quotes String* in order to distinguish the end of a *Single Quotes String*. A forward slash (`/`) is an escape character **only** after an apostrophe (`'`) in *Single Quotes String*. If `/` isn't in *Single Quotes String* or `/` isn't after `', /` is a normal character.

Pass **Full Value** format:

Bash

```
az some-command --contact "{name:'bill'/s',age:20,paid:true}"
```

Pass **Partial Value** format:

Bash

```
az some-command --contact name="'bill'/s'"
```

If **value** isn't in *Single Quotes String*, you don't need to add an escape character after `'`.

Pass **Partial Value** format:

Bash

```
az some-command --contact name="bill's"
```

Here's another example using the following JSON:

JSON

```
{
  "name": "Bill",
  "motto": "One man's bug is another man's lesson.",
  "age": 20,
  "paid": true,
  "emails": [
    "Bill@microsoft.com",
    "Bill@outlook.com"
  ]
}
```

In **Full Value** format, use a *Single Quotes String* and replace the `'` with `'/`.

Bash

```
az some-command --contact "{name:Bill,motto:'One man'/s bug is another
man'/s lesson.',age:20,paid:true,emails:
[Bill@microsoft.com,Bill@outlook.com]}"
```

In **Partial Value** format, a value containing an apostrophe can be parsed as string, surrounded by double quotes.

Bash

```
az some-command --contact motto="One man's bug is another man's lesson."
```

Work with NULL values

Pass the word "null" as a string value

Sometime you need to pass a "null" string value. In order to distinguish with `null` value, it needs to be a *Single Quotes String*. For example if you want to pass "null" string into the `name` property in the `--contact` parameter:

JSON:

JSON

```
{
  "name": "null",
  "age": 20,
  "paid": true
}
```

Use `Full Value` format:

Bash

```
az some-command --contact "{name:'null',age:20,paid:true}"
```

Use `Partial Value` format:

Bash

```
az some-command --contact name="'null'"
```

Pass a `null` value

Shorthand syntax support `null` keyword in both `Full Value` and `Partial Value` formats.

For example if you want to pass following object with a `null` value `address` property in the `--contact` parameter:

JSON

```
{  
  "name": "Bill",  
  "age": 20,  
  "paid": true,  
  "emails": [  
    "Bill@microsoft.com",  
    "Bill@outlook.com"  
,  
  "address": null  
}
```

Use `Full Value` format:

Bash

```
az some-command --contact "{name:Bill,age:20,paid:true,emails:[Bill@microsoft.com,Bill@outlook.com],address:null}"
```

Use `Partial Value` format:

Bash

```
az some-command --contact name=Bill address=null
```

Use a `null` value in update commands

In update commands, a `null` value is often used to unset properties of an object, or remove elements of an array or a dictionary.

JSON

```
{  
  "contact": {  
    "name": "Bill",  
    "age": 20,  
    "paid": true,  
    "emails": [  
      "Bill@microsoft.com",  
      "Bill@outlook.com"  
,  
    "address": {  
      "country": "USA",  
      "company": "Microsoft",  
      "details": {  
        "line1": "15590 NE 31st St",  
        "line2": "Redmond, WA"  
    }  
  }  
}
```

```
        }
    },
},
"other_properties": {}
}
```

If there already exists a resource with the JSON values shown above, passing a `null` value in an update command resets the key's value.

Bash

```
az some-update-command --contact address=null
```

Here's another example that removes the first element of a resource's email:

Bash

```
az some-update-command --emails [0]=null
```

Use `??` to show help

The use of double question marks `??` is a special keyword to show the available help for a parameter or subproperty. It can also be used within shorthand syntax to get help.

ⓘ Note

The Bash shell uses `?` as a wildcard. Make sure the double question marks `??` are wrapped in the **double quotes**.

Use `??` to show a parameter's help message

Show help message of `--contact` parameter:

Bash

```
az some-command --contact "??"
```

Use `??` in Full Value format

Show help message of `--contant` parameter when writing `Full Value`:

Bash

```
az some-command --contact "{???"
```

Bash

```
az some-command --contact "{name:Bill,???"
```

Show help message of `--contant.address` property when writing `Full Value`:

Bash

```
az some-command --contact "{name:Bill,address:???"
```

Show help message of `--contant.address.country` property when writing `Full Value`:

Bash

```
az some-command --contact "{name:Bill,address:{country:???"
```

Show help message of `--contant.emails` property when writing `Full Value`:

Bash

```
az some-command --contact "{name:Bill,address:{country:USA},emails:???"
```

Show help message of the element of `--contant.emails` property when writing `Full Value`:

Bash

```
az some-command --contact "{name:Bill,address:{country:USA},emails:[???"
```

Use `??` in Partial Value format

Show help message of `--contant.address` property when writing `Partial Value`:

Bash

```
az some-command --contact address="???"
```

Show help message of `--contant.emails` property when writing `Partial Value`:

Bash

```
az some-command --contact emails="???"
```

Show help message of the element of `--contant.emails` property when writing `Partial Value`:

Bash

```
az some-command --contact emails[0]="???"
```

See also

- [Quoting differences between scripting languages - JSON strings](#)

Use Microsoft Copilot for Azure to learn how to use the Azure CLI

Article • 08/28/2024

[Microsoft Copilot for Azure](#) (preview) is an AI-powered tool to help you do more with Azure. It unifies knowledge and data across hundreds of Azure services to increase productivity, reduce costs, and provide deep insights. Microsoft Copilot for Azure (preview) helps you learn about Azure by answering questions, and it provides information tailored to your own Azure resources and environment. By letting you express your goals in natural language, Copilot simplifies your Azure management experience. This benefits Azure CLI users because the knowledge of Azure CLI is built into Copilot!

Here are some of the benefits of using Microsoft Copilot for Azure:

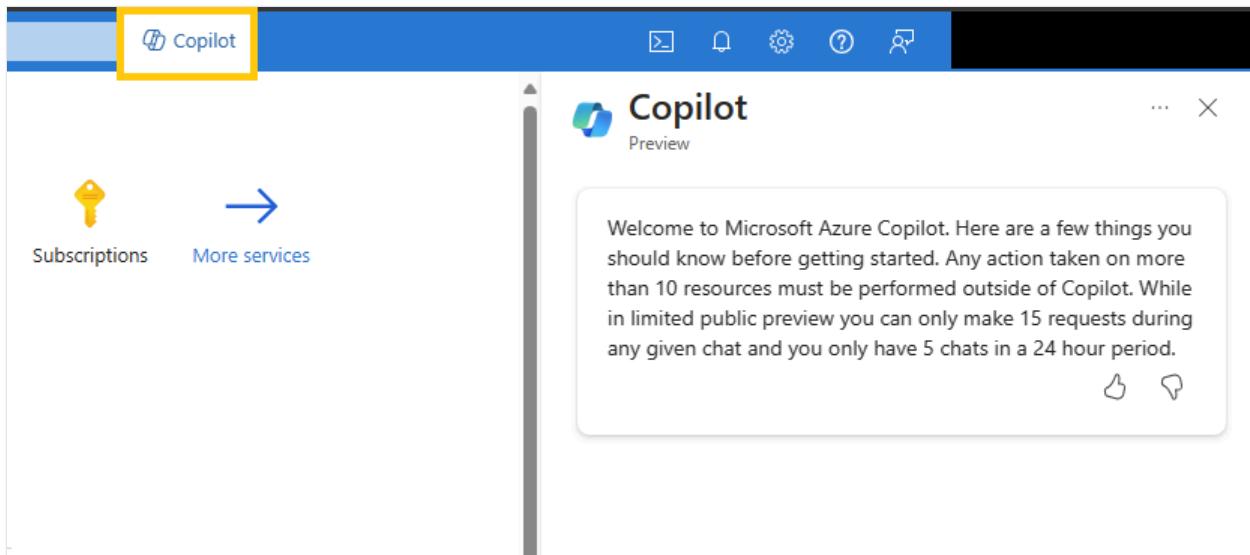
- **Design:** Create and configure the services needed and align with organizational policies.
- **Operate:** Answer questions, author complex commands, and manage resources.
- **Troubleshoot:** Orchestrate across Azure services for insights to summarize issues, identify causes, and suggest solutions.
- **Optimize:** Receive recommendations to improve costs, scalability, and reliability.

Enable Microsoft Copilot for Azure

To enable access to Microsoft Copilot for Azure (preview) for your organization, [complete the registration form](#). The application process only needs to be completed once per tenant. Check with your administrator if you have questions about joining the preview.

Open Microsoft Copilot for Azure

To use Microsoft Copilot, go to the Azure Portal and select the **Copilot** icon on the top taskbar. The copilot window will open with a chat assistant where you can ask questions.



When you tell Microsoft Copilot about a task you want to perform with Azure CLI, it provides a script with the necessary commands.

Search for reference commands

Ask Microsoft Copilot to assist with using the reference command `az group`. Note the placeholder values (marked in blue font) you need to update with actual values.



Copilot

Preview

...



I want to create a new virtual machine using Azure CLI

To create a new virtual machine using Azure CLI, you can follow these steps:

1. Create a resource group to contain the virtual machine:

▶ Run

```
az group create --name $resourceGroup --location  
$location
```

2. Create a virtual network with a subnet for the virtual machine:

▶ Run

```
az network vnet create --name $vnetName --  
resource-group $resourceGroup --subnet-name  
$subnetName --subnet-prefix $subnetPrefix
```

3. Create a subnet within the virtual network:

▶ Run

```
az network vnet subnet create --name $subnetName  
--resource-group $resourceGroup --vnet-name  
$vnetName --address-prefix $addressPrefix
```

Replace the variables (marked in blue) with actual values, or define the variables in your variable block before running the command.

Replace variables

Azure CLI

```
# Create a resource group  
az group create --name myResourceGroup --location eastus
```

Look up an Azure CLI script

Suppose you encounter an Azure CLI script that includes references and parameters that are new to you. Use Microsoft Copilot to look up the reference command by entering the command as if you were in a terminal:

What does this Azure CLI script do? az vm list --query "[].{objectID:id}" --output table

The Azure CLI script `az vm list --query "[].{objectID:id}" --output table` lists the object ID of all virtual machines in a table format. The `--query` parameter is used to customize the properties and columns you want to show in the list output, and the `[]` syntax is used to select all elements in the array.

The `table` output format prints output as an ASCII table, making it easy to read and scan. Some keys are not printed in the table view by default, but they can be included using the JMESPath re-keying feature to change the key name and avoid filtering.

For more information, you can refer to the [Output formats for Azure CLI commands](#) and [How to query Azure CLI command output using a JMESPath query](#) documentation.

AI-generated content may be incorrect



● 4 of 15 requests

Microsoft Copilot references and links other Microsoft Learn documents related to your specific CLI topic and automatically generates follow-up questions.

Search for a task or job to be done

Use Azure CLI with Microsoft Copilot to help you create an Azure SQL database.

How do I create an Azure SQL database with Azure CLI?

To create an Azure SQL database using Azure CLI, you can follow these steps:

1. Create an Azure SQL server:

```
az sql server create --name $serverName --resource-group $resourceGroup --location $location --admin-user $adminUser --admin-password $adminPassword
```

Replace `$serverName` with the name of your server, `$resourceGroup` with the name of your resource group, `$location` with the location of your server, `$adminUser` with your admin username, and `$adminPassword` with your admin password.

2. Create an Azure SQL database:

```
az sql db create --name $databaseName --resource-group $resourceGroup --server $serverName --service-objective $serviceObjective
```

Replace `$databaseName` with the name of your database, `$resourceGroup` with the name of your resource group, `$serverName` with the name of your server, and `$serviceObjective` with the service objective of your database.

Please replace the variables with your actual values.

AI-generated content may be incorrect



● 7 of 15 requests

Follow the steps provided:

1. Create an Azure SQL server:

Azure CLI

```
# Variable block
$serverName="myServerName"
$resourceGroup="myResourceGroup"
$locationName="eastus2"
$adminUser="myAdminUser"
$adminPassword="myPassword"

az sql server create --name $serverName --resource-group $resourceGroup --location $location --admin-user $adminUser --admin-password $adminPassword
```

2. Create your Azure SQL database:

Azure CLI

```
# Variable block
databaseName="myDatabaseName"
resourceGroup= "myResourceGroup"
serverName="myServerName"
serviceObjective="basic"

az sql db create --name $databaseName --resource-group $resourceGroup -
-server $serverName --service-objective $serviceObjective
```

Let us know how we did

Microsoft Copilot for Azure values your feedback. At the bottom of the Copilot response, give us a **thumbs up** if the Copilot response was helpful to you. Give us a **thumbs down** when we need to make improvements.

See also

- [Generate Azure CLI scripts using Microsoft Copilot for Azure \(preview\)](#)
- [Apply for access to Microsoft Copilot for Azure ↗](#)

Use the Azure REST API with Azure CLI

Article • 06/27/2024

Representational State Transfer (REST) APIs are service endpoints that support different sets of HTTP operations (or methods). These HTTP methods allow you to perform different actions for your service's resources. The `az rest` command should only be used when an existing [Azure CLI command](#) isn't available.

This article demonstrates the PUT, PATCH, GET, POST, and DELETE HTTP requests to manage Azure Container Registry resources. The [Azure Container Registry](#) is a managed registry service that allows you to create and maintain Azure container registries that store container images and related artifacts.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
A blue rectangular button with a white 'A' icon and the text 'Launch Cloud Shell'. To the right of the button is a small blue square icon with a white arrow pointing outwards.
A blue rectangular button with a white 'A' icon and the text 'Launch Cloud Shell'. To the right of the button is a small blue square icon with a white arrow pointing outwards.
- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

Tips for using `az rest`

Here's some helpful information when working with `az rest`:

- The `az rest` command automatically authenticates using the logged-in credential.
- If Authorization header is not set, it attaches header `Authorization: Bearer <token>`, where `<token>` is retrieved from [Microsoft Entra ID](#).

- The target resource of the token will be derived from the `--url` parameter when the `--url` parameter starts with an endpoint from the output of the `az cloud show --query endpoints` command. The `--url` parameter required.
- Use the `--resource` parameter for a custom resource.
- If Content-Type header is not set and `--body` is a valid JSON string, Content-Type header will default to "application/json".
- When using `--uri-parameters` for requests in the form of OData, make sure to escape `$` in different environments: in `Bash`, escape `$` as `\$` and in `PowerShell`, escape `$` as ``$`.

Use PUT to create an Azure Container Registry

Use the PUT HTTP method to create a new Azure Container Registry.

Azure CLI

```
# Command format example
az rest --method put \
  --url
https://management.azure.com/subscriptions/<subscriptionId>/resourceGroups/<
resourceGroup>/providers/Microsoft.ContainerRegistry/registries/<containerRe
gistryName>?api-version=2023-01-01-preview \
  --body "{'location': '<locationName>', 'sku': {'name': '<skuName>'}, 'properties': {'adminUserEnabled': '<propertyValue>'}}"
```

Here's an example with completed parameters:

Bash

Azure CLI

```
# Variable block
let "randomIdentifier=$RANDOM*$RANDOM"
subscriptionId="00000000-0000-0000-0000-000000000000"
resourceGroup="msdocs-app-rg$randomIdentifier"
containerRegistryName="msdocscr$randomIdentifier"
locationName="westus"
skuName="Standard"
propertyValue="true"

# Create resource group
az group create --name $resourceGroup --location $locationName --output
json

# Invoke request
az rest --method put \
```

```
--url
https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/
/$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName?api-version=2023-01-01-preview \
--body "{'location': '$locationName', 'sku': {'name': '$skuName'}, \
'properties': {'adminUserEnabled': '$propertyValue'}}"
```

JSON output for both Bash and Powershell:

JSON

```
{
  "id": "/subscriptions/<subscriptionId>/resourceGroups/<resourceGroup>/providers/Microsoft.ContainerRegistry/registries/<containerRegistryName>",
  "location": "<location>",
  "name": "<containerRegistryName>",
  "properties": {
    "adminUserEnabled": true,
    "anonymousPullEnabled": false,
    "creationDate": "2024-01-03T18:38:36.7089583Z",
    "dataEndpointEnabled": false,
    "dataEndpointHostNames": [],
    "encryption": {
      "status": "disabled"
    },
    "loginServer": "<containerRegistryName>.azurecr.io",
    "networkRuleBypassOptions": "AzureServices",
    "policies": {
      "azureADAuthenticationAsArmPolicy": {
        "status": "enabled"
      },
      "exportPolicy": {
        "status": "enabled"
      },
      "quarantinePolicy": {
        "status": "disabled"
      },
      "retentionPolicy": {
        "days": 7,
        "lastUpdatedTime": "2024-01-03T19:44:53.9770581+00:00",
        "status": "disabled"
      },
      "softDeletePolicy": {
        "lastUpdatedTime": "2024-01-03T19:44:53.9771117+00:00",
        "retentionDays": 7,
        "status": "disabled"
      },
      "trustPolicy": {
        "status": "disabled",
        "type": "Notary"
      }
    }
  }
}
```

```
        },
      },
      "privateEndpointConnections": [],
      "provisioningState": "Succeeded",
      "publicNetworkAccess": "Enabled",
      "zoneRedundancy": "Disabled"
    },
    "sku": {
      "name": "Standard",
      "tier": "Standard"
    },
    "systemData": {
      "createdAt": "2024-01-03T18:38:36.7089583+00:00",
      "createdBy": "<username>@microsoft.com",
      "createdByType": "User",
      "lastModifiedAt": "2024-01-03T19:44:53.684342+00:00",
      "lastModifiedBy": "<username>@microsoft.com",
      "lastModifiedByType": "User"
    },
    "tags":{},
    "type": "Microsoft.ContainerRegistry/registries"
  }
}
```

Use PATCH to update your Azure Container Registry

Update your Azure Container Registry by using the PATCH HTTP request. Edit the `--body` parameter with the properties you want to update. This example uses the variables set in the previous section, and updates the SKU name (`$skuName="Premium"`) of the Azure Container Registry.

Bash

Azure CLI

```
#Variable Block
$skuName="Premium"

az rest --method patch \
  --url
https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName?api-version=2023-01-01-preview \
  --body "{'location': '$locationName', 'sku': {'name': '$skuName'}, 'properties': {'adminUserEnabled': '$propertyValue'}}"
```

The following JSON dictionary output has fields omitted for brevity:

JSON

```
{  
  "id":  
    "/subscriptions/<subscriptionId>/resourceGroups/<resourceGroup>/providers/Microsoft.ContainerRegistry/registries/<containerRegistryName>",  
  "location": "westus",  
  "name": "<containerRegistryName>",  
  "properties": {...},  
  "sku": {  
    "name": "Premium",  
    "tier": "Premium"  
  },  
  "systemData": {...},  
  "type": "Microsoft.ContainerRegistry/registries"  
}
```

Use GET to retrieve your Azure Container Registry

Use the GET HTTP request see the update results from the PATCH request. This example uses the variables set in the previous section.

Bash

Azure CLI

```
az rest --method get \  
  --url  
  https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName?api-version=2023-01-01-preview
```

The output for GET method is the same as the one shown for PUT.

Use POST to regenerate your Azure Container Registry credentials

Use the POST HTTP request to regenerate one of the login credentials for the Azure Container Registry created in this article.

Bash

Azure CLI

```
# Variable block
$passwordValue="password"

az rest --method post \
  --url
https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/
$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName/regenerateCredential?api-version=2023-01-01-preview \
  --body "{ 'name': '$passwordValue' }"
```

The following JSON dictionary output has fields omitted for brevity:

JSON

```
{
  "passwords": [
    {
      "name": "password",
      "value": "<passwordValue>"
    },
    {
      "name": "password2",
      "value": "<passwordValue2>"
    }
  ],
  "username": "<containerRegistryName>"
}
```

After the request is complete, your specified Azure Container Registry credentials will be regenerated with a new password along with your existing password (password2).

Use DELETE to delete your Azure Container Registry

Use the DELETE HTTP request to delete an existing Azure Container Registry.

Bash

Azure CLI

```
az rest --method delete \
    --url
https://management.azure.com/subscriptions/$subscriptionId/resourceGroups/
$resourceGroup/providers/Microsoft.ContainerRegistry/registries/$containerRegistryName?api-version=2023-01-01-preview
```

Additional `az rest` example for Microsoft Graph

Sometimes it helps to see an example for a different scenario, so here is an example that uses the [Microsoft Graph API](#). To update redirect URLs for an [Application](#), call the [Update application REST API](#), as in this code:

Azure CLI

```
# Get the application
az rest --method GET \
    --uri 'https://graph.microsoft.com/v1.0/applications/b4e4d2ab-e2cb-45d5-
a31a-98eb3f364001'

# Update `redirectUris` for `web` property
az rest --method PATCH \
    --uri 'https://graph.microsoft.com/v1.0/applications/b4e4d2ab-e2cb-45d5-
a31a-98eb3f364001' \
    --body '{"web":{"redirectUris":["https://myapp.com"]}}'
```

Clean up resources

When you are finished with the resources created in this article, you can delete the resource group. When you delete the resource group, all resources in that resource group are deleted.

Azure CLI

```
az group delete --resource-group <resourceGroupName>
```

See also

- [Azure REST API reference](#)
- [az resource command](#)

 Collaborate with us on
GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Azure CLI interactive mode

Article • 03/08/2024

What is the Azure CLI interactive mode?

Azure CLI Interactive Mode (az interactive) provides users an interactive environment to run Azure CLI commands. The interactive mode makes it easier for you to learn the Azure CLI's capabilities, command syntax, and output formats. It provides autocompletion dropdowns, auto-cached suggestions combined with runtime documentation, and includes examples about how each command is used. Azure CLI Interactive Mode aims to provide an ideal experience for users learning to use Azure CLI commands.

Prerequisites

Use a Bash or PowerShell environment in [Azure Cloud Shell](#) or [run the Azure CLI locally](#) in PowerShell 7.

 [Launch Cloud Shell](#) .

Run Azure CLI interactive mode

Use Azure CLI in interactive mode by running the following command:

```
Azure CLI
az interactive
```

The Azure CLI interactive mode places you in an interactive shell with autocompletion, command descriptions, and examples.

```
Windows PowerShell * + - x
az>> webapp create
--name [Required] Name of the new web app. Web app name can contain only allow alphanumeric char^J..
--plan [Required] name or resource id of the app service plan. Use 'appservice plan create' to ge...
--resource-group [Required] Name of resource group. You can configure the default group using 'az configure...
--assign-identity accept system or user assigned identities separated by spaces. Use '[system]' to refer sys^J..
--deployment-container-image-name Container image name from Docker Hub, e.g. publisher/image-name:tag
--deployment-local-git enable local git
--deployment-source-branch the branch to deploy

Create a web app. *
[2] Create a web app with a Java 11 runtime.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "java:11:Java SE:11"
[3] Create a web app with a NodeJS 10.14 runtime and deployed from a local git repository.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "node:12LTS" --deployment-local-git
[4] Create a web app with an image from DockerHub.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i nginx
[5] Create a web app with an image from a private DockerHub registry.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i MyImageName -s username -w password

1/2 CTRL+Y (^) CTRL+N (v)

/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]        : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd]::[num]    : do a step by step tutorial of example
$              : get the exit code of the previous command
%[cmd]          : set a scope, and scopes can be chained with spaces
%..            : go back a scope
[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:
```

Configure your options in interactive mode

Interactive mode optionally displays command descriptions, parameter descriptions, and command examples.

Turn descriptions and examples on or off using **F1**.

```
Windows PowerShell * + - x
Do you want examples (y/n): y
Do you want scenarios recommendation (y/n): y
Please restart the interactive mode for changes to take effect.

az>>
az>>

Try [Space] or 'next' to get Command Recommendation *
```

```
[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:
```

You can turn the display of parameter defaults on or off using **F2**.

A new Recommender is added which can make the completion ability more intelligent and provide the scenario completion! If you want to disable this feature, you can use 'az config set interactive.enable_recommender=False' to disable it.

A command preload mechanism was added to prevent lagging and command run errors. You can skip preloading in a single pass by CTRL+C or turn it off by setting 'az config set interactive.enable_preloading=False'

Loading command table... Expected time around 1 minute.

az>>

Try [Space] or 'next' to get Command Recommendation *

No Default Values

```
/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]         : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd>::[num]    : do a step by step tutorial of example
$              : get the exit code of the previous command
%[cmd]         : set a scope, and scopes can be chained with spaces
%..            : go back a scope
```

[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:

You can toggle the display of some key gestures by using **F3**.

A new Recommender is added which can make the completion ability more intelligent and provide the scenario completion! If you want to disable this feature, you can use 'az config set interactive.enable_recommender=False' to disable it.

A command preload mechanism was added to prevent lagging and command run errors. You can skip preloading in a single pass by CTRL+C or turn it off by setting 'az config set interactive.enable_preloading=False'

Loading command table... Expected time around 1 minute.

az>>

Try [Space] or 'next' to get Command Recommendation *

```
/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]         : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd>::[num]    : do a step by step tutorial of example
$              : get the exit code of the previous command
%[cmd]         : set a scope, and scopes can be chained with spaces
%..            : go back a scope
```

[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:

Scope

In interactive mode, you can edit the scope to a specific command group to have all commands interpreted in that scope. This method provides an efficient way of inputting your commands, demonstrated in the following examples using the command group **vm**.

Instead of typing these commands:

Azure CLI

```
az>> vm create -n <myVirtualMachine> -g <myResourceGroup> --image Ubuntu2204
az>> vm list -o table
```

You can scope to the `vm` command group and type these commands:

Azure CLI

```
az>> %%vm
az vm>> create -n <myVirtualMachine> -g <myResourceGroup> --image Ubuntu2204
az vm>>list -o table
```

You can edit the scope to lower-level command groups as well like `vm image` by using `%%vm image`. In this case, since we're already scoped to `vm`, we would use `%%image`.

Azure CLI

```
az vm>> %%image
az vm image>>
```

At that point, we can edit the scope back up to `vm` using `%%%`, or we can edit the scope to the root with just `%`.

Azure CLI

```
az vm image>> %%
az>>
```

Query in interactive mode

You can execute a JMESPath query on the results of the last command that you executed by using `??` followed by a JMESPath query. For example, after you created a group, you can retrieve the `id` of the new group.

Azure CLI

```
az>> group create -n <myResourceGroup> -l westEurope
az>> "?? id"
```

You can also use this syntax to use the result of the previous command as an argument for your next command.* For instance, after having listed all groups, you can query for more specific results by using the list you just generated.

Azure CLI

```
az>> vm create --name <myVirtualMachine> --resource-group <myResourceGroup>
--image Ubuntu2204 --no-wait -o json
az>> group list -o json

# List all the resources of type `virtualMachine` on the first group whose
location is westeurope.
az>> resource list -g "?? [?location=='westeurope'].name | [0]" --query "[?
type=='Microsoft.Compute/virtualMachines'].name"
```

To learn more about querying the results of your commands, see [Query command results with the Azure CLI](#).

Run Bash and PowerShell commands in interactive mode

You can run both Bash and PowerShell commands without leaving interactive mode.

For Bash scripts, use the `#[cmd]` syntax:

Bash

```
az>> #dir
```

For PowerShell scripts, use the regular PowerShell syntax:

PowerShell

```
az>> $PSVersionTable
```

View examples of your commands in interactive mode

You can scroll to the next page of examples using `CTRL + N` and the previous page using `CTRL + Y`.

```
az>> webapp create
--name                [Required] Name of the new web app. Web app name can contain only allow alphanumeric char^J.. .
--plan                [Required] name or resource id of the app service plan. Use 'appservice plan create' to ge...
--resource-group      [Required] Name of resource group. You can configure the default group using 'az configur...
--assign-identity     accept system or user assigned identities separated by spaces. Use '[system]' to refer sys^J..
--deployment-container-image-name Container image name from Docker Hub, e.g. publisher/image-name:tag
--deployment-local-git enable local git
--deployment-source-branch the branch to deploy

Create a web app.          *

[2] Create a web app with a Java 11 runtime.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "java:11:Java SE:11"
[3] Create a web app with a NodeJS 10.14 runtime and deployed from a local git repository.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "node:12LTS" --deployment-local-git
[4] Create a web app with an image from DockerHub.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i nginx
[5] Create a web app with an image from a private DockerHub registry.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i MyImageName -s username -w password

1/2 CTRL+Y (^) CTRL+N (v)

/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]         : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd]::[num]    : do a step by step tutorial of example
$              : get the exit code of the previous command
%*[cmd]        : set a scope, and scopes can be chained with spaces
%..            : go back a scope
[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:
```

You can also look at a specific example using `::#`.

```
Azure CLI

az>> webapp create ::2

az>> webapp create ::2
(tutorial) az>> webapp create -g

webapp create -g MyResourceGroup -p MyPlan -n
MyUniqueAppName --runtime "java:11:Java SE:11"

In Tutorial Mode: Press [Enter] after typing each part
```

Artificial intelligence (AI) functionalities

The Azure CLI team has combined AI with Azure CLI interactive mode to provide users with a more beginner-friendly interactive experience. These AI functionalities allow users to learn about how to utilize Azure CLI interactive mode with more ease and efficiency.

The new AI features consist of:

- **Intelligent completion:** The AI-powered recommendation system uses telemetry analysis in order to predict and analyze customer behavior. These results lead to the prioritization of the most used and needed content options.
- **Command recommendation:** The AI offers subsequent commands based on the most used commands in other similar sessions.
- **Scenario identification:** The AI identifies the current possible scenarios and recommends a set of related command combinations for your specific end to end scenario.
- **Usable knowledge search:** The AI allows you to input natural language in order to search for commands for specific usage scenarios.

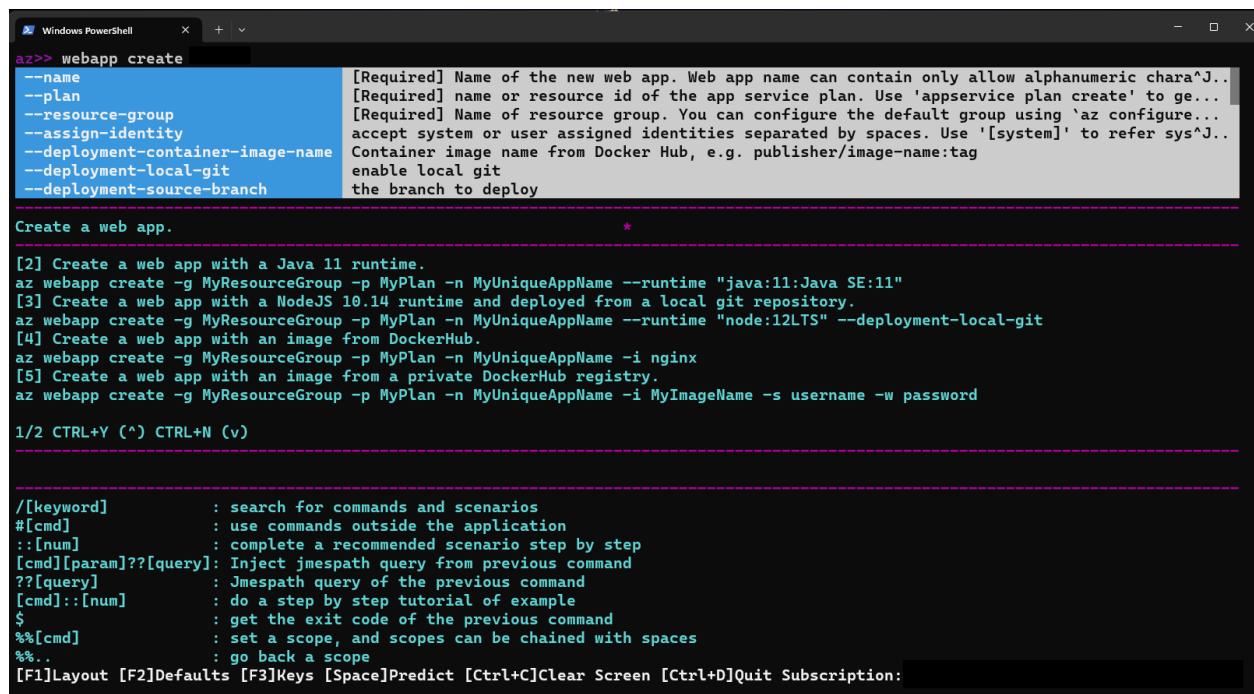
How to use AI functionalities with Azure CLI interactive mode

When using interactive mode within Azure CLI, upgrade to the latest version to use the new features.



```
Azure CLI
az extension add --name interactive --upgrade
```

To use **command recommendation**, run a command and click `space` or `next`.



```
Windows PowerShell
az>> webapp create
--name          [Required] Name of the new web app. Web app name can contain only allow alphanumeric chara^J..
--plan          [Required] name or resource id of the app service plan. Use 'appservice plan create' to ge...
--resource-group [Required] Name of resource group. You can configure the default group using 'az config...
--assign-identity [Required] accept system or user assigned identities separated by spaces. Use '[system]' to refer sys^J..
--deployment-container-image-name Container image name from Docker Hub, e.g. publisher/image-name:tag
--deployment-local-git    enable local git
--deployment-source-branch the branch to deploy

Create a web app. *
[2] Create a web app with a Java 11 runtime.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "java:11:Java SE:11"
[3] Create a web app with a NodeJS 10.14 runtime and deployed from a local git repository.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName --runtime "node:12LTS" --deployment-local-git
[4] Create a web app with an image from DockerHub.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i nginx
[5] Create a web app with an image from a private DockerHub registry.
az webapp create -g MyResourceGroup -p MyPlan -n MyUniqueAppName -i MyImageName -s username -w password

1/2 CTRL+Y (^) CTRL+N (v)

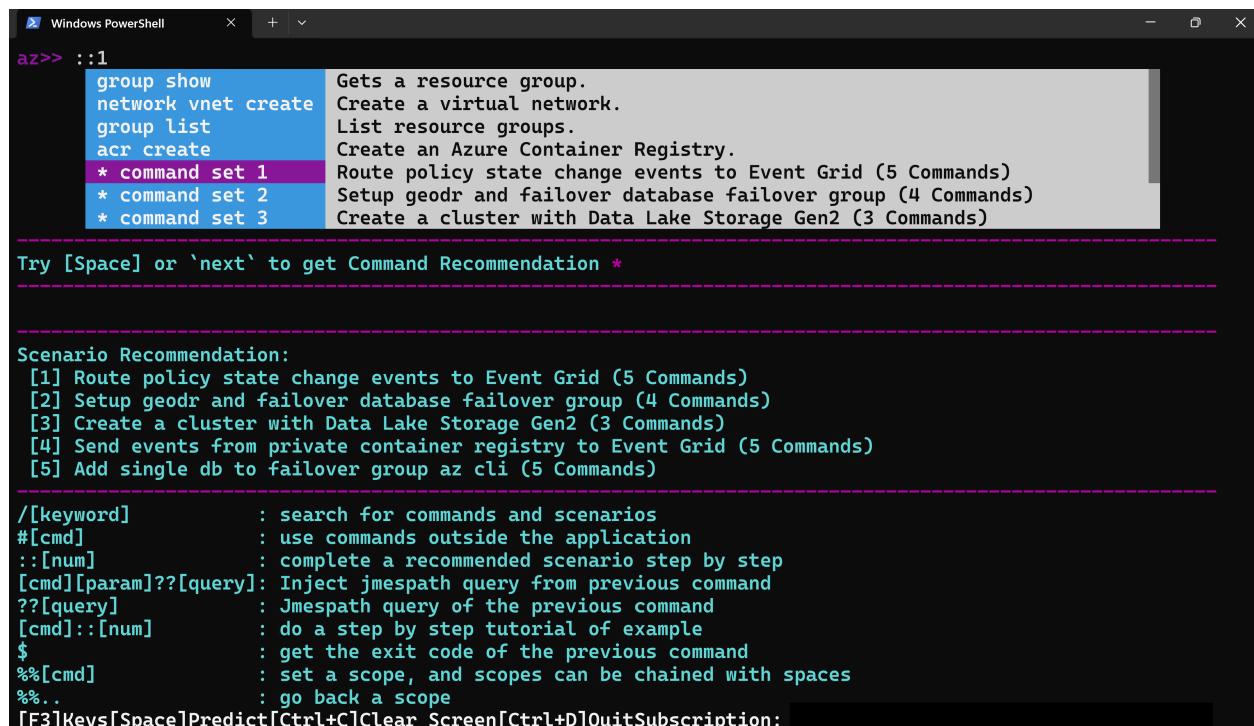
/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]         : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd]::[num]    : do a step by step tutorial of example
$              : get the exit code of the previous command
%[cmd]         : set a scope, and scopes can be chained with spaces
%..           : go back a scope
[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:
```

To use **scenario recommendation**, make sure you run a command first. Then after the command runs successfully, enter a `space`. Various command sets are listed, and you

will see that whichever command set you select, a `::[num]` is added after the `space` you entered.

ⓘ Note

For scenario identification, if you enter `::[num]` with no space before it, then it will return an error. Make sure to enter a single `space` in order to be able to view the list of command sets.



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The command entered is `az::1`. A tooltip-like box appears over the command, listing several command sets:

- `group show`: Gets a resource group.
- `network vnet create`: Create a virtual network.
- `group list`: List resource groups.
- `acr create`: Create an Azure Container Registry.
- * command set 1**: Route policy state change events to Event Grid (5 Commands)
- * command set 2**: Setup geodr and failover database failover group (4 Commands)
- * command set 3**: Create a cluster with Data Lake Storage Gen2 (3 Commands)

Below the tooltip, the text "Try [Space] or 'next' to get Command Recommendation *" is displayed. Further down, under "Scenario Recommendation:", a list of five items is shown:

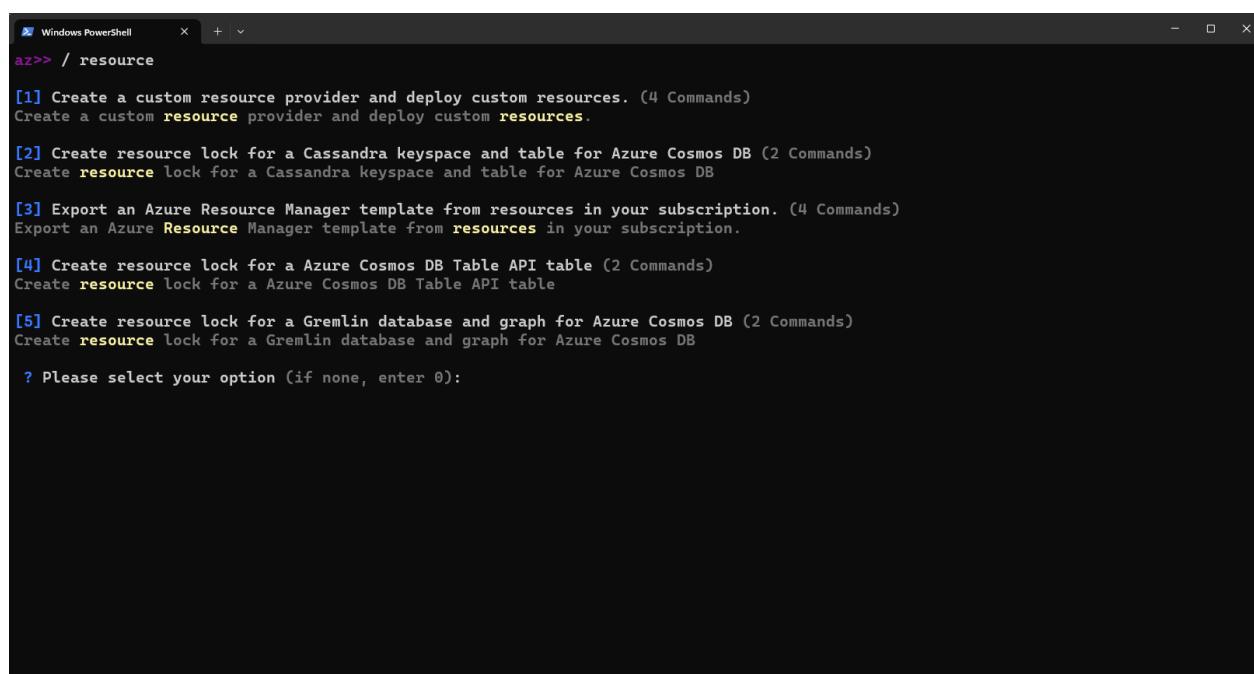
- [1] Route policy state change events to Event Grid (5 Commands)
- [2] Setup geodr and failover database failover group (4 Commands)
- [3] Create a cluster with Data Lake Storage Gen2 (3 Commands)
- [4] Send events from private container registry to Event Grid (5 Commands)
- [5] Add single db to failover group az cli (5 Commands)

The bottom of the window shows a list of keyboard shortcuts and help text:

- `/[keyword]` : search for commands and scenarios
- `#/[cmd]` : use commands outside the application
- `::[num]` : complete a recommended scenario step by step
- `[cmd][param]??[query]`: Inject jmespath query from previous command
- `??[query]` : Jmespath query of the previous command
- `[cmd]::[num]` : do a step by step tutorial of example
- `$` : get the exit code of the previous command
- `%*[cmd]` : set a scope, and scopes can be chained with spaces
- `%%` : go back a scope

At the bottom right, the text "[F3]Keys[Space]Predict[Ctrl+C]Clear Screen[Ctrl+D]QuitSubscription:" is visible.

To use **usable knowledge search**, run `/ [keyword]` to search for commands and scenarios.



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The command entered is `/ resource`. The output shows a list of five numbered steps:

- [1] Create a custom resource provider and deploy custom resources. (4 Commands)
Create a custom `resource` provider and deploy custom `resources`.
- [2] Create resource lock for a Cassandra keyspace and table for Azure Cosmos DB (2 Commands)
Create `resource` lock for a Cassandra keyspace and table for Azure Cosmos DB
- [3] Export an Azure Resource Manager template from resources in your subscription. (4 Commands)
Export an Azure `Resource` Manager template from `resources` in your subscription.
- [4] Create resource lock for a Azure Cosmos DB Table API table (2 Commands)
Create `resource` lock for a Azure Cosmos DB Table API table
- [5] Create resource lock for a Gremlin database and graph for Azure Cosmos DB (2 Commands)
Create `resource` lock for a Gremlin database and graph for Azure Cosmos DB

At the bottom, the text "? Please select your option (if none, enter 0):" is displayed.

To disable the new AI features for Azure CLI interactive mode input the following command:

```
Azure CLI
```

```
az config set interactive.enable_recommender=False
```

Azure cloud management with the Azure CLI

Article • 06/26/2023

If you work across different regions or use [Azure Stack](#), you may need to use more than one cloud. Microsoft provides clouds for compliance with regional laws, which are available for your use. This article shows you how to get information on clouds, change the current cloud, and register or unregister new clouds.

List available clouds

You can list available clouds with the `az cloud list` command. This command shows which cloud is currently active, what its current profile is, and information on regional suffixes and host names.

To get the active cloud and a list of all the available clouds:

```
Azure CLI
az cloud list --output table

Output
+-----+-----+-----+
| IsActive | Name | Profile |
+-----+-----+-----+
| True | AzureCloud | latest |
| | AzureChinaCloud | latest |
| | AzureUSGovernment | latest |
| | AzureGermanCloud | latest |
```

The currently active cloud has `True` in the `IsActive` column. Only one cloud can be active at any time. To get more detailed information on a cloud, including the endpoints that it uses for Azure services, use the `cloud show` command:

```
Azure CLI
az cloud show --name AzureChinaCloud --output json

JSON
```

```
{  
  "endpoints": {  
    "activeDirectory": "https://login.chinacloudapi.cn",  
    "activeDirectoryDataLakeResourceId": null,  
    "activeDirectoryGraphResourceId": "https://graph.chinacloudapi.cn/",  
    "activeDirectoryResourceId":  
      "https://management.core.chinacloudapi.cn/",  
      "batchResourceId": "https://batch.chinacloudapi.cn/",  
      "gallery": "https://gallery.chinacloudapi.cn/",  
      "management": "https://management.core.chinacloudapi.cn/",  
      "resourceManager": "https://management.chinacloudapi.cn",  
      "sqlManagement": "https://management.core.chinacloudapi.cn:8443/",  
      "vmImageAliasDoc": "https://raw.githubusercontent.com/Azure/azure-rest-api-specs/master/arm-compute/quickstart-templates/aliases.json"  
    },  
    "isActive": false,  
    "name": "AzureChinaCloud",  
    "profile": "latest",  
    "suffixes": {  
      "azureDatalakeAnalyticsCatalogAndJobEndpoint": null,  
      "azureDatalakeStoreFileSystemEndpoint": null,  
      "keyvaultDns": ".vault.azure.cn",  
      "sqlServerHostname": ".database.chinacloudapi.cn",  
      "storageEndpoint": "core.chinacloudapi.cn"  
    }  
}
```

Switch the active cloud

To set the default cloud using a configuration file, see [CLI configuration values and environment variables](#). To switch the active cloud, run the `az cloud set` command. This command takes one required argument, the name of the cloud.

Azure CLI

```
az cloud set --name AzureChinaCloud
```

Important

If your authentication for the activated cloud has expired, you need to re-authenticate before performing any other CLI tasks. If this is your first time switching to the new cloud, you also need to set the active subscription. For instructions on authenticating, see [Sign in with Azure CLI](#). For information on subscription management, see [Manage Azure subscriptions with Azure CLI](#)

Register a new cloud

Register a new cloud if you have your own endpoints for Azure Stack. Creating a cloud is done with the [az cloud register](#) command. This command requires a name and a set of service endpoints. To learn how to register a cloud for use with Azure Stack, see [Use API version profiles with Azure CLI in Azure Stack](#).

You don't need to register information for the China, US Government, or German regions. These clouds are managed by Microsoft and available by default. For more information on all of the available endpoint settings, see the [documentation for az cloud register](#).

Registering a cloud doesn't automatically switch to it. Use the `az cloud set` command to select the newly created cloud.

Update an existing cloud

If you have permissions, you can also update an existing cloud. Updating a cloud switches to a different Azure services profile or modifies the connection endpoints. Update a cloud with the [az cloud update](#) command, which takes the same arguments as `az cloud register`.

Unregister a cloud

If you no longer need a created cloud, it can be unregistered with the [az cloud unregister](#) command:

Azure CLI

```
az cloud unregister --name MyCloud
```

How to use variables in Azure CLI commands

Article • 07/26/2024

In addition to specifying values directly in a command, you can provide values in several ways:

- Use shell variables
- Set a subscription for use in multiple commands
- Create default values for some parameters

This article discusses various ways to specify values in Azure CLI commands.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
A blue rectangular button with a white 'A' icon on the left, the text 'Launch Cloud Shell' in the center, and a small blue square with a white arrow icon on the right.
- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).

Use shell variables

Azure CLI runs in a shell. This article uses Bash. For information about other scripting languages, see [Choose the right Azure command-line tool](#). You can use variables in Bash to pass values for parameters to commands. Using variables with the Azure CLI also allows reuse of commands, either piecemeal or in scripts.

This example creates a new storage disk of the same type as the storage disk on an existing virtual machine.

Bash

```
Azure CLI

# Bash script
# Assign values to variables
MyResourceGroup=ContosoRGforVM
MySubscription="Contoso subscription"
vmName=VM01

# Get a value for a variable based on an existing virtual machine
osType=$(az vm get-instance-view --resource-group $MyResourceGroup \
--name $vmName --subscription "$MySubscription" \
--query 'storageProfile.osDisk.osType' --output tsv)

# Create a disk of the same type by using the variable value
az disk create --resource-group $MyResourceGroup --name DestinationDisk \
--size-gb 20 --os-type $osType
```

This example shows how to assign values to variables that are reused, like `MyResourceGroup` and `osType`. The command `az vm get-instance-view` combined with the query `storageProfile.osDisk.osType` returns the disk's OS type. Wrapping the command with `$()` assigns the command's return value to `osType`. To learn more about `--query` and JMESPath queries see [How to query Azure CLI command output using a JMESPath query](#).

When you assign a value to a variable from another command, be sure that the command uses a compatible output format. The `az vm get-instance-view` command uses the `tsv` output format. This option returns values without extra formatting, keys, or other symbols. Some output formats include structure or characters like quotation marks. For more information, see [Output formats for Azure CLI commands](#).

In this example, the `MySubscription` variable must be in quotation marks. The value of the variable contains spaces, which the command can't parse. If you work only with subscription IDs, you don't need to use quotation marks.

Set a subscription

Many commands require a specific subscription. Azure resources exist in resource groups, which exist in subscriptions. Azure CLI uses a default subscription when you are

in a session. To see your current subscription value, run the [az account show](#) command:

```
Azure CLI
```

```
az account show --output table
```

You might have access to only one subscription. For more information, see [Use Azure subscriptions with Azure CLI](#). You can use the [az account set](#) command to set your current subscription:

```
Azure CLI
```

```
az account set --subscription "My Demos"
```

After you set your subscription, you can omit `--Subscription` parameter. For more information, see [Use Azure subscriptions with Azure CLI](#).

Create default values

You can set values for some parameters by using the [az config set](#) command. This example sets a default resource group:

```
Azure CLI
```

```
az config set defaults.group=ContosoRGforVM
```

After running this command, you can run the following command to create a storage account in the ContosoRGforVM resource group:

```
Azure CLI
```

```
az storage account create --name storage135 --location eastus --sku Standard_LRS
```

Notice that there's no resource group specified in the command. For more information, see [Set a default resource group](#).

Tip

Commands getting values for parameters in different ways can be confusing. If a command gives an unexpected result, such as not being able to find a resource group, there may be a default value.

If you encounter an error, run the command again with the parameter and value specified. An explicit value for a parameter always takes precedence over other options.

You can specify values for several parameters this way. For more information, see [Azure CLI configuration](#).

Clean up resources

If you created resources to try any of the commands in this article, you can remove them by using the [az group delete](#) command:

Azure CLI

```
az group delete --name ContosoRGforVM  
az group delete --name ContosoStorageRG
```

This command removes the group and all the resources that it contains at once.

See also

- [Learn to use Bash with the Azure CLI](#)
- [Tips to use the Azure CLI successfully](#)
- [How to query Azure CLI command output](#)

How-to work with global parameters in Azure CLI

Article • 08/28/2024

When you work with the Azure CLI reference commands, there are arguments that are available for most commands. These arguments are called "global parameters," as they're common to almost all Azure CLI reference commands.

--help global parameter

To get information about a reference group or command, use the `--help` global parameter. The `--help` parameter displays Azure CLI reference information in your console including available subgroup and parameter lists.

```
Azure CLI

az group create --name myResourceGroup --location eastus --help
```

The following console output shown here has fields omitted for brevity:

```
Output

Command
  az group create : Create a new resource group.

Arguments
  --location -l          [Required] : Location. Values from: `az
  account list-locations`.
                                         You can configure the default
  location using `az
                                         configure --defaults
  location=<location>`.

  --name --resource-group -g -n [Required] : Name of the new resource
  group.

  --managed-by             : The ID of the resource that
  manages this resource
                                         group.

  --tags                  : Space-separated tags:
  key[=value] [key[=value] ...].
                                         Use "" to clear existing
  tags.

Global Arguments
  --debug                 : Increase logging verbosity to
  show all debug logs.
```

```
--help -h : Show this help message and  
exit.  
--only-show-errors : Only show errors, suppressing  
warnings.
```

Examples

```
Create a new resource group in the West US region.  
az group create -l westus -n MyResourceGroup
```

--verbose global parameter

Learn more in-depth details about logging by adding the `--verbose` global parameter to your reference command. Verbose output is helpful when trying to understand why certain commands take longer than others to execute.

Azure CLI

```
az group create --name myResourceGroup --location eastus --verbose
```

The following console output shown here has fields omitted for brevity, but note the additional information on the execution time.

Output

```
{...}  
Command ran in 1.413 seconds (init: 0.487, invoke: 0.926)
```

--debug global parameter

If the extra details from the `--verbose` parameter aren't enough and you wish to view the entire log, use the `--debug` global parameter. Also provide the output generated with `--debug` when submitting a bug report.

Azure CLI

```
az group create --name myrResourceGroup --location eastus --debug
```

The following console output shown here has fields omitted for brevity:

Output

```
cli.knack.cli: Command arguments: ['group', 'create', '--name',
```

```
'myResourceGroup', '--location', 'eastus', '--debug']
cli.knack.cli: __init__ debug log:
[...]
cli.knack.cli: Event: CommandInvoker.OnTransformResult [<function
_resource_group_transform at 0x000001D9B72123B0>, <function
_x509_from_base64_to_hex_transform at 0x000001D9B7212440>]
cli.knack.cli: Event: CommandInvoker.OnFilterResult []
{...}
[...]
telemetry.process: Return from creating process
telemetry.main: Finish creating telemetry upload process.
```

--only-show-errors global parameter

During the process of debugging, remove the warnings returned by a reference command by using the `--only-show-errors` global parameter.

Without `--only-show-errors`

Azure CLI

`az interactive`

Output console:

Output

This command is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus

Any comments or concerns? You can use the 'feedback' command! We would greatly appreciate it.

A new Recommender is added which can make the completion ability more intelligent and provide the scenario completion!

If you want to disable this feature, you can use 'az config set interactive.enable_recommender=False' to disable it.

A command preload mechanism was added to prevent lagging and command run errors.

You can skip preloading in a single pass by CTRL+C or turn it off by setting 'az config set interactive.enable_preloading=False'

Loading command table... Expected time around 1 minute.

--subscription global parameter

You can use the `--subscription` global parameter to set a specific subscription for a command. The following example changes the active subscription using the subscription ID:

```
Azure CLI
```

```
az account set --subscription "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
```

--output global parameter

To show the output of a command in a specific format, specify the `--output` global parameter. In this example, the output is shown as a table:

ⓘ Note

The accepted values for output are: json, jsonc, none, table, tsv, yaml, and yamlc. The default output value is json.

```
Azure CLI
```

```
az vm list --resource-group myResourceGroup --output table
```

Output console:

```
Output
```

Name	ResourceGroup	Location
-----	-----	-----
myVm	myResourceGroup	eastus

To learn more about the different output formats, see [Output formats for Azure CLI commands](#).

--query global parameter

There are times the output console might display extensive amounts of information that you need to filter through. To do so, use the `--query` global parameter to execute a

JMESPath query [↗](#) on the results of commands. JMESPath is a query language for JSON, which gives you the ability to select and modify data from Azure CLI output.

This example returns the SSH public keys authorized to connect to the virtual machine by adding `--query`:

Azure CLI

```
az vm show --resource-group myResourceGroup --name myVm --query  
"osProfile_linuxConfiguration.ssh.publicKeys"
```

Output console:

Output

```
[  
  {  
    "keyData": "ssh-rsa  
AAAAB3NzaC1yc2EAAAQABAAQ...  
  }  
]
```

To learn more about querying command output, see [How to query Azure CLI command output using a JMESPath query](#).

See also

- [Azure CLI configuration](#)
- [Use the Azure CLI successfully](#)

Output formats for Azure CLI commands

Article • 06/26/2024

The Azure CLI uses JSON as its default output format, but offers other formats. Use the `--output` (`--out` or `-o`) parameter to format CLI output. The argument values and types of output are:

 Expand table

<code>--output</code>	Description
<code>json</code>	JSON string. This setting is the default
<code>jsonc</code>	Colorized JSON
<code>table</code>	ASCII table with keys as column headings
<code>tsv</code>	Tab-separated values, with no keys
<code>yaml</code>	YAML, a human-readable alternative to JSON
<code>yamlc</code>	Colorized YAML
<code>none</code>	No output other than errors and warnings

⚠ Warning

Use an output format of `none` or store command output in a variable to avoid exposing secrets such as API keys and credentials. **Note:** Certain CI/CD environments may store the output of the executed commands in logs. It is a good practice to confirm what is written in those log files and who has access to the logs. For more information, see [None output format](#).

JSON output format (default)

The following example displays the list of virtual machines in your subscriptions in the default JSON format.

Azure CLI

```
az vm list --output json
```

The following output has some fields omitted for brevity, and identifying information replaced.

JSON

```
[  
  {  
    "availabilitySet": null,  
    "diagnosticsProfile": null,  
    "hardwareProfile": {  
      "vmSize": "Standard_DS1"  
    },  
    "id":  
      "/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtualMachines/DemoVM010",  
    "instanceView": null,  
    "licenseType": null,  
    "location": "westus",  
    "name": "DemoVM010",  
    "networkProfile": {  
      "networkInterfaces": [  
        {  
          "id":  
            "/subscriptions/.../resourceGroups/demorg1/providers/Microsoft.Network/networkInterfaces/DemoVM010VMNic",  
          "primary": null,  
          "resourceGroup": "demorg1"  
        }  
      ]  
    },  
    ...  
    ...  
    ...  
  }]
```

YAML output format

The `yaml` format prints output as [YAML](#), a plain-text data serialization format. YAML tends to be easier to read than JSON, and easily maps to that format. Some applications and CLI commands take YAML as configuration input, instead of JSON.

Azure CLI

```
az vm list --output yaml
```

The following output has some fields omitted for brevity, and identifying information replaced.

YAML

```
- availabilitySet: null
  diagnosticsProfile: null
  hardwareProfile:
    vmSize: Standard_DS1_v2
  id: /subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtualMachines/DemoVM010
  identity: null
  instanceView: null
  licenseType: null
  location: westus
  name: ExampleVM1
  networkProfile:
    networkInterfaces:
      - id: /subscriptions/.../resourceGroups/DemoRG1/providers/Microsoft.Network/networkInterfaces/DemoVM010Nic
        primary: null
        resourceGroup: DemoRG1
  ...
  ...

```

Table output format

The `table` format prints output as an ASCII table, making it easy to read and scan.

Nested objects aren't included in table output, but can still be filtered as part of a query. Some fields aren't included in the table, so this format is best when you want a quick, human-searchable overview of data.

Azure CLI

```
az vm list --output table
```

Output

Name	ResourceGroup	Location
DemoVM010	DEMORG1	westus
demovm212	DEMORG1	westus
demovm213	DEMORG1	westus
KBDemo001VM	RGDEMO001	westus
KBDemo020	RGDEMO001	westus

You can use the `--query` parameter to customize the properties and columns you want to show in the list output. The following example shows how to select just the VM Name and the Resource Group Name in the `list` command.

Azure CLI

```
az vm list --query "[].{resource:resourceGroup, name:name}" --output table
```

Output

Resource	Name
DEMORG1	DemoVM010
DEMORG1	demovm212
DEMORG1	demovm213
RGDEMO001	KBDemo001VM
RGDEMO001	KBDemo020

ⓘ Note

Some keys are not printed in the table view by default. These are `id`, `type`, and `etag`. If you need to see these in your output, you can use the JMESPath re-keying feature to change the key name and avoid filtering.

Azure CLI

```
az vm list --query "[].{objectId:id}" --output table
```

For more about using queries to filter data, see [Use JMESPath queries with Azure CLI](#).

TSV output format

The `tsv` output format returns tab- and newline-separated values without extra formatting, keys, or other symbols. This format makes it easy to consume the output into other commands and tools that need to process the text in some form. Like the `table` format, `tsv` doesn't print nested objects.

Using the preceding example with the `tsv` option outputs the tab-separated result.

Azure CLI

```
az vm list --output tsv
```

Output

```
None    None  
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtua  
lMachines/DemoVM010    None    None    westus    DemoVM010        None  
Succeeded    DEMORG1    None        Microsoft.Compute/virtualMachines  
cbd56d9b-9340-44bc-a722-25f15b578444  
None    None  
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtua  
lMachines/demovm212    None    None    westus    demovm212        None  
Succeeded    DEMORG1    None        Microsoft.Compute/virtualMachines  
4bdac85d-c2f7-410f-9907-ca7921d930b4  
None    None  
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtua  
lMachines/demovm213    None    None    westus    demovm213        None  
Succeeded    DEMORG1    None        Microsoft.Compute/virtualMachines  
2131c664-221a-4b7f-9653-f6d542fbfa34  
None    None  
/subscriptions/.../resourceGroups/RGDEMO001/providers/Microsoft.Compute/virt  
ualMachines/KBDemo001VM    None    None    westus    KBDemo001VM  
None    Succeeded    RGDEMO001    None  
Microsoft.Compute/virtualMachines    14e74761-c17e-4530-a7be-9e4ff06ea74b  
None    None  
/subscriptions/.../resourceGroups/RGDEMO001/providers/Microsoft.Compute/virt  
ualMachines/KBDemo020    None    None    westus    KBDemo020        None  
Succeeded    RGDEMO001    None        Microsoft.Compute/virtualMachines  
36baa9-9b80-48a8-b4a9-854c7a858ece
```

One restriction of the TSV output format is that there isn't a guarantee on output ordering. The CLI makes a best effort to preserve ordering by sorting keys in the response JSON alphabetically, and then printing their values in order for TSV output. There is no guarantee that the order is always identical, since the Azure service response format can change.

In order to enforce consistent ordering, you'll need to use the `--query` parameter and the [multiselect list](#) format. When a CLI command returns a single JSON dictionary, use the general format `[key1, key2, ..., keyN]` to force a key order. For CLI commands that return an array, use the general format `[].[key1, key2, ..., keyN]` to order column values.

For example, to order the information displayed above by ID, location, resource group, and VM name:

```
az vm list --output tsv --query '[].[id, location, resourceGroup, name]'
```

Output

```
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtualMachines/DemoVM010    westus    DEMORG1    DemoVM010
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtualMachines/demovm212    westus    DEMORG1    demovm212
/subscriptions/.../resourceGroups/DEMORG1/providers/Microsoft.Compute/virtualMachines/demovm213    westus    DEMORG1    demovm213
/subscriptions/.../resourceGroups/RGDEMO001/providers/Microsoft.Compute/virtualMachines/KBDemo001VM    westus    RGDEMO001    KBDemo001VM
/subscriptions/.../resourceGroups/RGDEMO001/providers/Microsoft.Compute/virtualMachines/KBDemo020    westus    RGDEMO001    KBDemo020
```

The next example shows how `tsv` output can be piped to other commands in bash. The query is used to filter output and force ordering, `grep` selects items that have text "RGD" in them, then the `cut` command selects the fourth field to show the name of the VM in output.

Azure CLI

```
az vm list --output tsv --query '[].[id, location, resourceGroup, name]' |
grep RGD | cut -f4
```

Output

```
KBDemo001VM
KBDemo020
```

The `tsv` output format is often used when assigning values to variables. This example gets the active subscription ID and stores it into a variable for use in a script.

Bash

Azure CLI

```
# Bash Script
subscriptionID=$(az account show --query id --output tsv)
echo "Using subscription ID $subscriptionID"
```

For more `--query` parameter examples, see [How to query Azure CLI command output](#).

None output format

Some Azure CLI commands output information you must protect. Here are four examples:

- passwords
- connection strings
- secrets
- keys

To protect secrets and keys when using Azure CLI commands, choose one of these options:

[\[+\] Expand table](#)

Option	Benefit	Use case
--output none output format	Keeps sensitive information from being displayed in your console. If your command fails, you'll still receive error messages.	1. Use when command output <i>can be</i> retrieved at a later time. 2. Use when you have no need for output.
		3. A common choice when a managed identity or a service principal is being used to manage Azure resources.
--query parameter	Stores output in a variable.	1. Use when command output <i>can't be</i> retrieved at a later time. 2. Use when you need to use a command output value in a script.

Use `none` and retrieve security information at a later time

Some Azure secrets can be retrieved at a later time. A good example is secrets stored in Azure Key Vault. In this example, create an Azure Key Vault secret using [az keyvault secret set](#) with the `--output none` option. You can retrieve the secret later using the [az keyvault secret show](#) command.

```
az keyvault secret set --name MySecretName \  
    --vault-name MyKeyVaultName \  
    --value MySecretValue\  
    --output none
```

Use `--query` and return security information to a variable

The use of `--query` to store output in a variable is technically not an output format. It is a solution to protect secrets, and is an alternative to using `--output none`. For example, when you reset a service principal credential, the password can't be retrieved again.

Reset a service principal credential returning output in the default json format:

Azure CLI

```
# reset service principal credentials using default output format (json).  
az ad sp credential reset --id myServicePrincipalID --output json
```

Console output showing the new password in the console.

Output

```
{  
    "appId": "myServicePrincipalID",  
    "password": "myServicePrincipalNewPassword",  
    "tenant": "myTenantID"  
}
```

A better solution is to return sensitive information to a variable.

Bash

Azure CLI

```
# Bash Script  
# reset service principal credentials returning results to a variable  
myNewPassword=$(az ad sp credential reset --id myServicePrincipalID --  
query password --output tsv)  
  
# Display the new password (remove this line in production for security)  
echo "New password: $myNewPassword"
```

For more examples on storing output to a variable, see [Use the Azure CLI successfully - pass values to another command](#). To learn more about `--query` parameter syntax, see [How to query Azure CLI command output](#).

Set the default output format

Azure CLI commands provide output that can be controlled in two ways:

[] [Expand table](#)

Output control	Benefit	How-to
Global setting	Select a default output value that you use the most so you don't have to continually provide an <code>--output</code> parameter for each reference command.	Specify a default output format using az config set .
Command parameter	Specify output at the command level and give your scripts maximum flexibility. You control console output and variable input for each reference command.	Override the default setting using a reference command's <code>--output</code> parameter.

The default output for the Azure CLI is `json`. Set the default output to `none` when console output isn't needed.

Azure CLI

```
az config set core.output=none
```

You can overwrite the default output of any Azure CLI reference command by using the `--output` parameter. Here's a script of commands that alter and test command output:

Azure CLI

```
# set your default output to table
az config set core.output=table

# show your active subscription in table format
# notice how only a subset of properties are returned in the table
az account show

# override your table default and show your active subscription in jsonc
# format
az account show --output jsonc
```

```
# reset your default output to json  
az config set core.output=json
```

See also

- [Azure CLI configuration](#)
- [How to query Azure CLI command output](#)
- [Tips to use the Azure CLI successfully](#)

How to use Azure Resource Manager (ARM) deployment templates with Azure CLI

Article • 09/26/2024

This article explains how to use Azure CLI with Azure Resource Manager templates (ARM templates) to deploy your resources to Azure. If you aren't familiar with the concepts of deploying and managing your Azure solutions, see [template deployment overview](#).

The deployment commands changed in Azure CLI version 2.2.0. The examples in this article require [Azure CLI version 2.20.0 or later](#).

To run this sample, install the latest version of the [Azure CLI](#). To start, run `az login` to create a connection with Azure.

Samples for the Azure CLI are written for the `bash` shell. To run this sample in Windows PowerShell or Command Prompt, you may need to change elements of the script.

If you don't have Azure CLI installed, you can use Azure Cloud Shell. For more information, see [Deploy ARM templates from Azure Cloud Shell](#).

Tip

We recommend [Bicep](#) because it offers the same capabilities as ARM templates and the syntax is easier to use. To learn more, see [How to deploy resources with Bicep and Azure CLI](#).

Required permissions

To deploy a Bicep file or ARM template, you need write access on the resources you're deploying and access to all operations on the Microsoft.Resources/deployments resource type. For example, to deploy a virtual machine, you need `Microsoft.Compute/virtualMachines/write` and `Microsoft.Resources/deployments/*` permissions. The what-if operation has the same permission requirements.

For a list of roles and permissions, see [Azure built-in roles](#).

Deployment scope

You can target your Azure deployment template to a resource group, subscription, management group, or tenant. Depending on the scope of the deployment, you use different commands.

- To deploy to a **resource group**, use [az deployment group create](#):

```
Azure CLI
```

```
az deployment group create --resource-group <resource-group-name> --  
template-file <path-to-template>
```

- To deploy to a **subscription**, use [az deployment sub create](#):

```
Azure CLI
```

```
az deployment sub create --location <location> --template-file <path-  
to-template>
```

For more information about subscription level deployments, see [Create resource groups and resources at the subscription level](#).

- To deploy to a **management group**, use [az deployment mg create](#):

```
Azure CLI
```

```
az deployment mg create --location <location> --template-file <path-to-  
template>
```

For more information about management group level deployments, see [Create resources at the management group level](#).

- To deploy to a **tenant**, use [az deployment tenant create](#):

```
Azure CLI
```

```
az deployment tenant create --location <location> --template-file  
<path-to-template>
```

For more information about tenant level deployments, see [Create resources at the tenant level](#).

For every scope, the user deploying the template must have the required permissions to create resources.

Deploy local template

You can deploy an ARM template from your local machine or one that is stored externally. This section describes deploying a local template.

If you're deploying to a resource group that doesn't exist, create the resource group. The name of the resource group can only include alphanumeric characters, periods, underscores, hyphens, and parenthesis. It can be up to 90 characters. The name can't end in a period.

Azure CLI

```
az group create --name ExampleGroup --location "Central US"
```

To deploy a local template, use the `--template-file` parameter in the deployment command. The following example also shows how to set a parameter value.

Azure CLI

```
az deployment group create \
  --name ExampleDeployment \
  --resource-group ExampleGroup \
  --template-file <path-to-template> \
  --parameters storageAccountType=Standard_GRS
```

The value of the `--template-file` parameter must be a Bicep file or a `.json` or `.jsonc` file. The `.jsonc` file extension indicates the file can contain `//` style comments. The ARM system accepts `//` comments in `.json` files. It does not care about the file extension. For more details about comments and metadata see [Understand the structure and syntax of ARM templates](#).

The Azure deployment template can take a few minutes to complete. When it finishes, you see a message that includes the result:

Output

```
"provisioningState": "Succeeded",
```

Deploy remote template

Instead of storing ARM templates on your local machine, you might prefer to store them in an external location. You can store templates in a source control repository (such as

GitHub). Or, you can store them in an Azure storage account for shared access in your organization.

ⓘ Note

To deploy a template or reference a linked template that is stored in a private GitHub repo, see a custom solution documented in [Creating a Custom and Secure Azure Portal Offering](#). You can create an [Azure function](#) that pulls the GitHub token out of Azure Key Vault.

If you're deploying to a resource group that doesn't exist, create the resource group. The name of the resource group can only include alphanumeric characters, periods, underscores, hyphens, and parenthesis. It can be up to 90 characters. The name can't end in a period.

Azure CLI

```
az group create --name ExampleGroup --location "Central US"
```

To deploy an external template, use the `template-uri` parameter.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--template-uri "https://raw.githubusercontent.com/Azure/azure-quickstart-templates/master/quickstarts/microsoft.storage/storage-account-create/azuredeploy.json" \
--parameters storageAccountType=Standard_GRS
```

The preceding example requires a publicly accessible URI for the template, which works for most scenarios because your template shouldn't include sensitive data. If you need to specify sensitive data (like an admin password), pass that value as a secure parameter. However, if you want to manage access to the template, consider using [template specs](#).

To deploy remote linked templates with relative path that are stored in a storage account, use `query-string` to specify the SAS token:

Azure CLI

```
az deployment group create \
--name linkedTemplateWithRelativePath \
--resource-group myResourceGroup \
```

```
--template-uri "https://stage20210126.blob.core.windows.net/template-staging/mainTemplate.json" \
--query-string $sasToken
```

For more information, see [Use relative path for linked templates](#).

Azure deployment template name

When deploying an ARM template, you can give the Azure deployment template a name. This name can help you retrieve the deployment from the deployment history. If you don't provide a name for the deployment, the name of the template file is used. For example, if you deploy a template named `azuredeploy.json` and don't specify a deployment name, the deployment is named `azuredeploy`.

Every time you run a deployment, an entry is added to the resource group's deployment history with the deployment name. If you run another deployment and give it the same name, the earlier entry is replaced with the current deployment. If you want to maintain unique entries in the deployment history, give each deployment a unique name.

To create a unique name, you can assign a random number.

Azure CLI

```
deploymentName='ExampleDeployment'$RANDOM
```

Or, add a date value.

Azure CLI

```
deploymentName='ExampleDeployment'$(date +"%d-%b-%Y")
```

If you run concurrent deployments to the same resource group with the same deployment name, only the last deployment is completed. Any deployments with the same name that haven't finished are replaced by the last deployment. For example, if you run a deployment named `newStorage` that deploys a storage account named `storage1`, and at the same time run another deployment named `newStorage` that deploys a storage account named `storage2`, you deploy only one storage account. The resulting storage account is named `storage2`.

However, if you run a deployment named `newStorage` that deploys a storage account named `storage1`, and immediately after it completes you run another deployment named `newStorage` that deploys a storage account named `storage2`, then you have two

storage accounts. One is named `storage1`, and the other is named `storage2`. But, you only have one entry in the deployment history.

When you specify a unique name for each deployment, you can run them concurrently without conflict. If you run a deployment named `newStorage1` that deploys a storage account named `storage1`, and at the same time run another deployment named `newStorage2` that deploys a storage account named `storage2`, then you have two storage accounts and two entries in the deployment history.

To avoid conflicts with concurrent deployments and to ensure unique entries in the deployment history, give each deployment a unique name.

Deploy template spec

Instead of deploying a local or remote template, you can create a [template spec](#). The template spec is a resource in your Azure subscription that contains an ARM template. It makes it easy to securely share the template with users in your organization. You use Azure role-based access control (Azure RBAC) to grant access to the template spec. This feature is currently in preview.

The following examples show how to create and deploy a template spec.

First, create the template spec by providing the ARM template.

```
Azure CLI  
  
az ts create \  
  --name storageSpec \  
  --version "1.0" \  
  --resource-group templateSpecRG \  
  --location "westus2" \  
  --template-file "./mainTemplate.json"
```

Then, get the ID for template spec and deploy it.

```
Azure CLI  
  
id = $(az ts show --name storageSpec --resource-group templateSpecRG --  
version "1.0" --query "id")  
  
az deployment group create \  
  --resource-group demoRG \  
  --template-spec $id
```

For more information, see [Azure Resource Manager template specs](#).

Preview changes

Before deploying your ARM template, you can preview the changes the template makes to your environment. Use the [what-if operation](#) to verify that the template makes the changes that you expect. What-if also validates the template for errors.

Parameters

To pass parameter values, you can use either inline parameters or a parameters file. The parameter file can be either a [Bicep parameters file](#) or a [JSON parameters file](#).

Inline parameters

To pass inline parameters, provide the values in `parameters`. For example, to pass a string and array to a template in a Bash shell, use:

Azure CLI

```
az deployment group create \
--resource-group testgroup \
--template-file <path-to-template> \
--parameters exampleString='inline string' exampleArray='("value1",
"value2")'
```

If you're using Azure CLI with Windows Command Prompt (CMD) or PowerShell, pass the array in the format: `exampleArray="['value1', 'value2']"`.

You can also get the contents of file and provide that content as an inline parameter.

Azure CLI

```
az deployment group create \
--resource-group testgroup \
--template-file <path-to-template> \
--parameters exampleString=@stringContent.txt
exampleArray=@arrayContent.json
```

Getting a parameter value from a file is helpful when you need to provide configuration values. For example, you can provide [cloud-init values for a Linux virtual machine](#).

The `arrayContent.json` format is:

JSON

```
[  
  "value1",  
  "value2"  
]
```

To pass in an object, for example, to set tags, use JSON. For example, your template might include a parameter like this one:

JSON

```
"resourceTags": {  
  "type": "object",  
  "defaultValue": {  
    "Cost Center": "IT Department"  
  }  
}
```

In this case, you can pass in a JSON string to set the parameter as shown in the following Bash script:

Azure CLI

```
tags='{"Owner": "Contoso", "Cost Center": "2345-324"}'  
az deployment group create --name addstorage --resource-group  
myResourceGroup \  
--template-file $templateFile \  
--parameters resourceName=abcdef4556 resourceTags="$tags"
```

Use double quotes around the JSON that you want to pass into the object.

You can use a variable to contain the parameter values. In Bash, set the variable to all of the parameter values and add it to the deployment command.

Azure CLI

```
params="prefix=start suffix=end"  
  
az deployment group create \  
--resource-group testgroup \  
--template-file <path-to-template> \  
--parameters $params
```

However, if you're using Azure CLI with Windows Command Prompt (CMD) or PowerShell, set the variable to a JSON string. Escape the quotation marks: `$params = '{"prefix": {"value": "start"}, "suffix": {"value": "end"}'}`.

JSON parameter files

Rather than passing parameters as inline values in your script, you might find it easier to use a parameters file, either a `.bicepparam` file or a JSON parameters file, that contains the parameter values. The parameters file must be a local file. External parameters files aren't supported with Azure CLI.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--template-file storage.json \
--parameters 'storage.parameters.json'
```

For more information about the parameter file, see [Create Resource Manager parameter file](#).

Bicep parameter files

With Azure CLI version 2.53.0 or later, and Bicep CLI version 0.22.6 or later, you can deploy a Bicep file by utilizing a Bicep parameter file. With the `using` statement within the Bicep parameters file, there is no need to provide the `--template-file` switch when specifying a Bicep parameter file for the `--parameters` switch. Including the `--template-file` switch results in an "Only a .bicep template is allowed with a .bicepparam file" error.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--parameters storage.bicepparam
```

The parameters file must be a local file. External parameters files aren't supported with Azure CLI. For more information about the parameters file, see [Create Resource Manager parameters file](#).

Comments and the extended JSON format

You can include `//` style comments in your parameter file, but you must name the file with a `.jsonc` extension.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--template-file storage.json \
--parameters '@storage.parameters.jsonc'
```

For more details about comments and metadata, see [Understand the structure and syntax of ARM templates](#).

If you are using Azure CLI with version 2.3.0 or older, you can deploy a template with multi-line strings or comments using the `--handle-extended-json-format` switch. For example:

JSON

```
{
  "type": "Microsoft.Compute/virtualMachines",
  "apiVersion": "2018-10-01",
  "name": "[variables('vmName')]", // to customize name, change it in
variables
  "location": "[
    parameters('location')
  ]", //defaults to resource group location
  /*
    storage account and network interface
    must be deployed first
  */
  "dependsOn": [
    "[resourceId('Microsoft.Storage/storageAccounts/',
variables('storageAccountName'))]",
    "[resourceId('Microsoft.Network/networkInterfaces/',
variables('nicName'))]"
  ],
}
```

Next steps

- To roll back to a successful deployment when you get an error, see [Rollback on error to successful deployment](#).
- To specify how to handle resources that exist in the resource group but aren't defined in the template, see [Azure Resource Manager deployment modes](#).
- To understand how to define parameters in your template, see [Understand the structure and syntax of ARM templates](#).
- For tips on resolving common deployment errors, see [Troubleshoot common Azure deployment errors with Azure Resource Manager](#).

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

How to deploy private ARM template with SAS token

Article • 03/20/2024

When your Azure Resource Manager template (ARM template) is located in a storage account, you can restrict access to the template to avoid exposing it publicly. You access a secured template by creating a shared access signature (SAS) token for the template, and providing that token during deployment. This article explains how to use Azure PowerShell or Azure CLI to securely deploy an ARM template with a SAS token.

You will find information on how to protect and manage access to your private ARM templates with directions on how to do the following:

- Create storage account with secured container
- Upload template to storage account
- Provide SAS token during deployment

ⓘ Important

Instead of securing your private template with a SAS token, consider using [template specs](#). With template specs, you can share your templates with other users in your organization and manage access to the templates through Azure RBAC.

Create storage account with secured container

The following script creates a storage account and container with public access turned off for template security.

PowerShell

Azure PowerShell

```
New-AzResourceGroup`  
  -Name ExampleGroup`  
  -Location "Central US"  
New-AzStorageAccount`  
  -ResourceGroupName ExampleGroup`  
  -Name {your-unique-name}`  
  -Type Standard_LRS`  
  -Location "Central US"
```

```
Set-AzCurrentStorageAccount ` 
-ResourceGroupName ExampleGroup ` 
-Name {your-unique-name} 
New-AzStorageContainer ` 
-Name templates ` 
-Permission Off
```

Upload private template to storage account

Now, you're ready to upload your template to the storage account. Provide the path to the template you want to use.

PowerShell

Azure PowerShell

```
Set-AzStorageBlobContent ` 
-Container templates ` 
-File c:\Templates\azuredeploy.json
```

Provide SAS token during deployment

To deploy a private template in a storage account, generate a SAS token and include it in the URI for the template. Set the expiry time to allow enough time to complete the deployment.

ⓘ Important

The blob containing the private template is accessible to only the account owner. However, when you create a SAS token for the blob, the blob is accessible to anyone with that URI. If another user intercepts the URI, that user is able to access the template. A SAS token is a good way of limiting access to your templates, but you should not include sensitive data like passwords directly in the template.

PowerShell

Azure PowerShell

```
# get the URI with the SAS token 
$templateuri = New-AzStorageBlobSASToken `
```

```
-Container templates `
-Blob azuredeploy.json `
-Permission r `
-ExpiryTime (Get-Date).AddHours(2.0) -FullUri

# provide URI with SAS token during deployment
New-AzResourceGroupDeployment `
-ResourceGroupName ExampleGroup `
-TemplateUri $templateuri
```

For an example of using a SAS token with linked templates, see [Using linked templates with Azure Resource Manager](#).

Next steps

- For an introduction to deploying templates, see [Deploy resources with ARM templates and Azure PowerShell](#).
- To define parameters in template, see [Authoring templates](#).

How to deploy resources with Bicep and Azure CLI

Article • 03/22/2024

This article explains how to use Azure CLI with Bicep files to deploy your resources to Azure. If you aren't familiar with the concepts of deploying and managing your Azure solutions, see [Bicep overview](#).

Prerequisites

You need a Bicep file to deploy. The file must be local.

You need Azure CLI and to be connected to Azure:

- **Install Azure CLI commands on your local computer.** To deploy Bicep files, you need [Azure CLI](#) version 2.20.0 or later.
- **Connect to Azure by using az login.** If you have multiple Azure subscriptions, you might also need to run [az account set](#).

Samples for the Azure CLI are written for the `bash` shell. To run this sample in Windows PowerShell or Command Prompt, you might need to change elements of the script.

If you don't have Azure CLI installed, you can use Azure Cloud Shell. For more information, see [Deploy Bicep files from Azure Cloud Shell](#).

Required permissions

To deploy a Bicep file or ARM template, you need write access on the resources you're deploying and access to all operations on the Microsoft.Resources/deployments resource type. For example, to deploy a virtual machine, you need

`Microsoft.Compute/virtualMachines/write` and `Microsoft.Resources/deployments/*` permissions. The what-if operation has the same permission requirements.

For a list of roles and permissions, see [Azure built-in roles](#).

Deployment scope

You can target your deployment to a resource group, subscription, management group, or tenant. Depending on the scope of the deployment, you use different commands.

- To deploy to a **resource group**, use [az deployment group create](#):

```
Azure CLI
```

```
az deployment group create --resource-group <resource-group-name> --  
template-file <path-to-bicep>
```

- To deploy to a **subscription**, use [az deployment sub create](#):

```
Azure CLI
```

```
az deployment sub create --location <location> --template-file <path-  
to-bicep>
```

For more information about subscription level deployments, see [Create resource groups and resources at the subscription level](#).

- To deploy to a **management group**, use [az deployment mg create](#):

```
Azure CLI
```

```
az deployment mg create --location <location> --template-file <path-to-  
bicep>
```

For more information about management group level deployments, see [Create resources at the management group level](#).

- To deploy to a **tenant**, use [az deployment tenant create](#):

```
Azure CLI
```

```
az deployment tenant create --location <location> --template-file  
<path-to-bicep>
```

For more information about tenant level deployments, see [Create resources at the tenant level](#).

For every scope, the user deploying the Bicep file must have the required permissions to create resources.

Deploy local Bicep file

You can deploy a Bicep file from your local machine or one that is stored externally. This section describes deploying a local Bicep file.

If you're deploying to a resource group that doesn't exist, create the resource group. The name of the resource group can only include alphanumeric characters, periods, underscores, hyphens, and parenthesis. It can be up to 90 characters. The name can't end in a period.

Azure CLI

```
az group create --name ExampleGroup --location "Central US"
```

To deploy a local Bicep file, use the `--template-file` switch in the deployment command. The following example also shows how to set a parameter value.

Azure CLI

```
az deployment group create \
  --name ExampleDeployment \
  --resource-group ExampleGroup \
  --template-file <path-to-bicep> \
  --parameters storageAccountType=Standard_GRS
```

The deployment can take a few minutes to complete. When it finishes, you see a message that includes the result:

Output

```
"provisioningState": "Succeeded",
```

Deploy remote Bicep file

Currently, Azure CLI doesn't support deploying remote Bicep files. You can use [Bicep CLI](#) to [build](#) the Bicep file to a JSON template, and then load the JSON file to the remote location. For more information, see [Deploy remote ARM JSON templates](#).

Parameters

To pass parameter values, you can use either inline parameters or a parameters file. The parameters file can be either a [Bicep parameters file](#) or a [JSON parameters file](#).

Inline parameters

To pass inline parameters, provide the values in `parameters`. For example, to pass a string and array to a Bicep file in a Bash shell, use:

Azure CLI

```
az deployment group create \
--resource-group testgroup \
--template-file <path-to-bicep> \
--parameters exampleString='inline string' exampleArray='["value1",
"value2"]'
```

If you're using Azure CLI with Windows Command Prompt (CMD) or PowerShell, pass the array in the format: `exampleArray="['value1','value2']"`.

You can also get the contents of file and provide that content as an inline parameter. Preface the file name with `@`.

Azure CLI

```
az deployment group create \
--resource-group testgroup \
--template-file <path-to-bicep> \
--parameters exampleString=@stringContent.txt
exampleArray=@arrayContent.json
```

Getting a parameter value from a file is helpful when you need to provide configuration values. For example, you can provide [cloud-init values for a Linux virtual machine](#).

The `arrayContent.json` format is:

JSON

```
[  
  "value1",  
  "value2"  
]
```

To pass in an object, for example, to set tags, use JSON. For example, your Bicep file might include a parameter like this one:

JSON

```
"resourceTags": {  
  "type": "object",  
  "defaultValue": {  
    "Cost Center": "IT Department"
```

```
}
```

In this case, you can pass in a JSON string to set the parameter as shown in the following Bash script:

Azure CLI

```
tags='{"Owner":"Contoso","Cost Center":"2345-324"}'  
az deployment group create --name addstorage --resource-group  
myResourceGroup \  
--template-file $bicepFile \  
--parameters resourceName=abcdef4556 resourceTags="$tags"
```

Use double quotes around the JSON that you want to pass into the object.

If you're using Azure CLI with Windows Command Prompt (CMD) or PowerShell, pass the object in the following format:

Azure CLI

```
$tags='{'Owner':'Contoso','Cost Center':'2345-324'}'  
az deployment group create --name addstorage --resource-group  
myResourceGroup \  
--template-file $bicepFile \  
--parameters resourceName=abcdef4556 resourceTags=$tags
```

You can use a variable to contain the parameter values. In Bash, set the variable to all of the parameter values and add it to the deployment command.

Azure CLI

```
params="prefix=start suffix=end"  
  
az deployment group create \  
--resource-group testgroup \  
--template-file <path-to-bicep> \  
--parameters $params
```

However, if you're using Azure CLI with Windows Command Prompt (CMD) or PowerShell, set the variable to a JSON string. Escape the quotation marks: `$params = '{"prefix": {"value": "start"}, "suffix": {"value": "end"} }'`.

The evaluation of parameters follows a sequential order, meaning that if a value is assigned multiple times, only the last assigned value is used. To ensure proper parameter assignment, it's advised to provide your parameters file initially and

selectively override specific parameters using the *KEY=VALUE* syntax. It's important to mention that if you're supplying a `bicepparam` parameters file, you can use this argument only once.

Bicep parameter files

Rather than passing parameters as inline values in your script, you might find it easier to use a parameters file, either a [Bicep parameters file](#) or a [JSON parameters file](#) that contains the parameter values. The parameters file must be a local file. External parameters files aren't supported with Azure CLI. For more information about the parameters file, see [Create Resource Manager parameters file](#).

With Azure CLI version 2.53.0 or later, and [Bicep CLI version 0.22.X or higher](#), you can deploy a Bicep file by utilizing a Bicep parameter file. With the `using` statement within the Bicep parameters file, there's no need to provide the `--template-file` switch when specifying a Bicep parameter file for the `--parameters` switch. Including the `--template-file` switch will result in an "Only a .bicep template is allowed with a .bicepparam file" error.

The following example shows a parameters file named *storage.bicepparam*. The file is in the same directory where the command is run.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--parameters storage.bicepparam
```

JSON parameter files

The following example shows a parameters file named *storage.parameters.json*. The file is in the same directory where the command is run.

Azure CLI

```
az deployment group create \
--name ExampleDeployment \
--resource-group ExampleGroup \
--template-file storage.bicep \
--parameters '@storage.parameters.json'
```

For more information about the parameters file, see [Create Resource Manager parameters file](#).

You can use inline parameters and a location parameters file in the same deployment operation. For more information, see [Parameter precedence](#).

Preview changes

Before deploying your Bicep file, you can preview the changes the Bicep file will make to your environment. Use the [what-if operation](#) to verify that the Bicep file makes the changes that you expect. What-if also validates the Bicep file for errors.

Deploy template specs

Currently, Azure CLI doesn't support creating template specs by providing Bicep files. However you can create a Bicep file with the [Microsoft.Resources/templateSpecs](#) resource to deploy a template spec. The [Create template spec sample](#) shows how to create a template spec in a Bicep file. You can also build your Bicep file to JSON by using the Bicep CLI, and then create a template spec with the JSON template.

Deployment name

When deploying a Bicep file, you can give the deployment a name. This name can help you retrieve the deployment from the deployment history. If you don't provide a name for the deployment, the name of the Bicep file is used. For example, if you deploy a Bicep file named `main.bicep` and don't specify a deployment name, the deployment is named `main`.

Every time you run a deployment, an entry is added to the resource group's deployment history with the deployment name. If you run another deployment and give it the same name, the earlier entry is replaced with the current deployment. If you want to maintain unique entries in the deployment history, give each deployment a unique name.

To create a unique name, you can assign a random number.

Azure CLI

```
deploymentName='ExampleDeployment'$RANDOM
```

Or, add a date value.

Azure CLI

```
deploymentName='ExampleDeployment'$(date +"%d-%b-%Y")
```

If you run concurrent deployments to the same resource group with the same deployment name, only the last deployment is completed. Any deployments with the same name that haven't finished are replaced by the last deployment. For example, if you run a deployment named `newStorage` that deploys a storage account named `storage1`, and at the same time run another deployment named `newStorage` that deploys a storage account named `storage2`, you deploy only one storage account. The resulting storage account is named `storage2`.

However, if you run a deployment named `newStorage` that deploys a storage account named `storage1`, and immediately after it completes you run another deployment named `newStorage` that deploys a storage account named `storage2`, then you have two storage accounts. One is named `storage1`, and the other is named `storage2`. But, you only have one entry in the deployment history.

When you specify a unique name for each deployment, you can run them concurrently without conflict. If you run a deployment named `newStorage1` that deploys a storage account named `storage1`, and at the same time run another deployment named `newStorage2` that deploys a storage account named `storage2`, then you have two storage accounts and two entries in the deployment history.

To avoid conflicts with concurrent deployments and to ensure unique entries in the deployment history, give each deployment a unique name.

Next steps

- To understand how to define parameters in your file, see [Understand the structure and syntax of Bicep files](#).

Use Azure Key Vault to pass secure parameter value during Bicep deployment

Article • 05/06/2024

Instead of putting a secure value (like a password) directly in your Bicep file or parameters file, you can retrieve the value from an [Azure Key Vault](#) during a deployment. When a `module` expects a `string` parameter with `secure:true` modifier, you can use the `getSecret` function to obtain a key vault secret. The value is never exposed because you only reference its key vault ID.

ⓘ Important

This article focuses on how to pass a sensitive value as a template parameter. When the secret is passed as a parameter, the key vault can exist in a different subscription than the resource group you're deploying to.

This article doesn't cover how to set a virtual machine property to a certificate's URL in a key vault. For a quickstart template of that scenario, see [Install a certificate from Azure Key Vault on a Virtual Machine ↗](#).

Deploy key vaults and secrets

To access a key vault during Bicep deployment, set `enabledForTemplateDeployment` on the key vault to `true`.

If you already have a key vault, make sure it allows template deployments.

Azure CLI

Azure CLI

```
az keyvault update --name ExampleVault --enabled-for-template-deployment true
```

To create a new key vault and add a secret, use:

Azure CLI

Azure CLI

```
az group create --name ExampleGroup --location centralus
az keyvault create \
    --name ExampleVault \
    --resource-group ExampleGroup \
    --location centralus \
    --enabled-for-template-deployment true
az keyvault secret set --vault-name ExampleVault --name
"ExamplePassword" --value "hVFkk965BuUv"
```

As the owner of the key vault, you automatically have access to create secrets. If the user working with secrets isn't the owner of the key vault, grant access with:

Azure CLI

Azure CLI

```
az keyvault set-policy \
    --upn <user-principal-name> \
    --name ExampleVault \
    --secret-permissions set delete get list
```

For more information about creating key vaults and adding secrets, see:

- [Set and retrieve a secret by using CLI](#)
- [Set and retrieve a secret by using PowerShell](#)
- [Set and retrieve a secret by using the portal](#)
- [Set and retrieve a secret by using .NET](#)
- [Set and retrieve a secret by using Node.js](#)

Grant access to the secrets

The user who deploys the Bicep file must have the

`Microsoft.KeyVault/vaults/deploy/action` permission for the scope of the resource group and key vault. The [Owner](#) and [Contributor](#) roles both grant this access. If you created the key vault, you're the owner and have the permission.

The following procedure shows how to create a role with the minimum permission, and how to assign the user.

1. Create a custom role definition JSON file:

```
JSON

{
  "Name": "Key Vault Bicep deployment operator",
  "IsCustom": true,
  "Description": "Lets you deploy a Bicep file with the access to the secrets in the Key Vault.",
  "Actions": [
    "Microsoft.KeyVault/vaults/deploy/action"
  ],
  "NotActions": [],
  "DataActions": [],
  "NotDataActions": [],
  "AssignableScopes": [
    "/subscriptions/00000000-0000-0000-0000-000000000000"
  ]
}
```

Replace "00000000-0000-0000-0000-000000000000" with the subscription ID.

2. Create the new role using the JSON file:

Azure CLI

Azure CLI

```
az role definition create --role-definition "<path-to-role-file>"
az role assignment create \
  --role "Key Vault Bicep deployment operator" \
  --scope /subscriptions/<Subscription-id>/resourceGroups/<resource-group-name> \
  --assignee <user-principal-name>
```

The samples assign the custom role to the user on the resource group level.

When using a key vault with the Bicep file for a [Managed Application](#), you must grant access to the **Appliance Resource Provider** service principal. For more information, see [Access Key Vault secret when deploying Azure Managed Applications](#).

Retrieve secrets in Bicep file

You can use the [getSecret function](#) in Bicep files to obtain a key vault secret. Note that the `getSecret` function is exclusively applicable to a `Microsoft.KeyVault/vaults`

resource. Additionally, it's restricted to usage within the `params` section of a module and can only be used with parameters with the `@secure()` decorator.

Another function called `az.getSecret()` function can be used in Bicep parameter files to retrieve key vault secrets. For more information, see [Retrieve secrets in parameters file](#).

Because the `getSecret` function can only be used in the `params` section of a module.

Let's create a `sql.bicep` in the same directory as the `main.bicep` file with the following content:

```
Bicep

param sqlServerName string
param location string = resourceGroup().location
param adminLogin string

@secure()
param adminPassword string

resource sqlServer 'Microsoft.Sql/servers@2023-08-01-preview' = {
    name: sqlServerName
    location: location
    properties: {
        administratorLogin: adminLogin
        administratorLoginPassword: adminPassword
        version: '12.0'
    }
}
```

Notice in the preceding Bicep file, the `adminPassword` parameter has a `@secure()` decorator.

The following Bicep file consumes the `sql.bicep` as a module. The Bicep file references an existing key vault, and calls the `getSecret` function to retrieve the key vault secret, and then passes the value as a parameter to the module.

```
Bicep

param sqlServerName string
param adminLogin string

param subscriptionId string
param kvResourceGroup string
param kvName string

resource kv 'Microsoft.KeyVault/vaults@2023-07-01' existing = {
    name: kvName
    scope: resourceGroup(subscriptionId, kvResourceGroup )
}
```

```
module sql './sql.bicep' = {
    name: 'deploySQL'
    params: {
        sqlServerName: sqlServerName
        adminLogin: adminLogin
        adminPassword: kv.getSecret('vmAdminPassword')
    }
}
```

Retrieve secrets in parameters file

If you don't want to use a module, you can retrieve key vault secrets in parameters file. However, the approach varies depending on whether you're using a JSON parameter file or a Bicep parameter file.

The following Bicep file deploys a SQL server that includes an administrator password. The password parameter is set to a secure string. But the Bicep doesn't specify where that value comes from.

Bicep

```
param sqlServerName string
param location string = resourceGroup().location
param adminLogin string

@secure()
param adminPassword string

resource sqlServer 'Microsoft.Sql/servers@2023-08-01-preview' = {
    name: sqlServerName
    location: location
    properties: {
        administratorLogin: adminLogin
        administratorLoginPassword: adminPassword
        version: '12.0'
    }
}
```

Now, create a parameters file for the preceding Bicep file.

Bicep parameter file

`az.getSecret` function can be used in a `.bicepparam` file to retrieve the value of a secret from a key vault.

Bicep

```
using './main.bicep'

param sqlServerName = '<your-server-name>'
param adminLogin = '<your-admin-login>'
param adminPassword = az.getSecret('<subscription-id>', '<rg-name>', '<key-vault-name>', '<secret-name>', '<secret-version>')
```

JSON parameter file

In the JSON parameters file, specify a parameter that matches the name of the parameter in the Bicep file. For the parameter value, reference the secret from the key vault. You reference the secret by passing the resource identifier of the key vault and the name of the secret:

In the following parameters file, the key vault secret must already exist, and you provide a static value for its resource ID.

JSON

```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "adminLogin": {
            "value": "<your-admin-login>"
        },
        "adminPassword": {
            "reference": {
                "keyVault": {
                    "id": "/subscriptions/<subscription-id>/resourceGroups/<rg-name>/providers/Microsoft.KeyVault/vaults/<key-vault-name>"
                },
                "secretName": "ExamplePassword"
            }
        },
        "sqlServerName": {
            "value": "<your-server-name>"
        }
    }
}
```

If you need to use a version of the secret other than the current version, include the `secretVersion` property.

JSON

```
"secretName": "ExamplePassword",  
"secretVersion": "cd91b2b7e10e492ebb870a6ee0591b68"
```

Next steps

- For general information about key vaults, see [What is Azure Key Vault?](#)
- For complete examples of referencing key secrets, see [key vault examples](#) on GitHub.
- For a Learn module that covers passing a secure value from a key vault, see [Manage complex cloud deployments by using advanced ARM template features](#).

View deployment history with Azure Resource Manager

Article • 03/20/2024

Azure Resource Manager enables you to view your deployment history. You can examine specific operations in past deployments and see which resources were deployed. This history contains information about any errors.

The deployment history for a resource group is limited to 800 deployments. As you near the limit, deployments are automatically deleted from the history. For more information, see [Automatic deletions from deployment history](#).

For help with resolving particular deployment errors, see [Troubleshoot common Azure deployment errors](#).

Correlation ID and support

Each deployment has a correlation ID, which is used to track related events. If you [create an Azure support request](#), support may ask you for the correlation ID. Support uses the correlation ID to identify the operations for the failed deployment.

The examples in this article show how to retrieve the correlation ID.

Resource group deployments

You can view details about a resource group deployment through the Azure portal, PowerShell, Azure CLI, or REST API.

Portal

1. Select the **resource group** you want to examine.

The screenshot shows the Microsoft Azure Resource Groups page. At the top, there's a search bar and navigation links. Below it, a table lists one record: 'ExampleGroup'. The 'Name' column has a checkbox next to 'ExampleGroup', which is highlighted with a red box. The 'Subscription' column shows 'Documentation Testing 1'. There are also filter and sorting options at the top of the table.

2. Select the link under Deployments.

The screenshot shows the Microsoft Azure Resource Group 'ExampleGroup' details page. On the left is a navigation menu with 'Overview', 'Activity log', 'Access control (IAM)', and 'Tags'. On the right, under the 'Essentials' section, there are fields for 'Subscription (Move)', 'Documentation', 'Subscription ID' (showing '11111111-1111-1111-1111-111111111111'), 'Location' (showing 'West US'), and a 'Deployments' section. The 'Deployments' section shows '1 Succeeded' and is highlighted with a red box.

3. Select one of the deployments from the deployment history.

The screenshot shows the Microsoft Azure Resource Group 'ExampleGroup' Deployment history page. On the left, a navigation menu includes 'Deployments', 'Security', and 'Policies'. The main area displays a table with columns for 'Deployment name' and 'Status'. A row for 'ExampleDeployment' is selected, with its status shown as 'Succeeded' and the entire row is highlighted with a red box.

4. A summary of the deployment is displayed, including the correlation ID.

ExampleDeployment | Overview

Deployment

» [Delete](#) [Cancel](#) [Redeploy](#) [Refresh](#)

Your deployment is complete

Deployment name: ExampleDeployment
Subscription: Documentation
Resource group: ExampleGroup

Start time: 11/9/2021, 1:56:14 PM
Correlation ID: 11111111-1111-1111-1111-111111111111

Deployment details (Download)

Resource	Type	Status	Operation details
storage	Microsoft.Storage/storageAccounts	OK	Operation details

Next steps

[Go to resource](#)

Subscription deployments

You can view the history of deployments to a subscription.

Portal

1. Select the **subscription** you want to examine.

Microsoft Azure

Home >

Subscriptions

View list of subscriptions for which you have role-based access control (RBAC) permissions to manage Azure resources. To Showing subscriptions in Microsoft directory. Don't see a subscription? [Switch directories](#)

My role [\(i\)](#)
8 selected

[Apply](#)

Showing 2 of 55 subscriptions Show only subscriptions selected in the [global subscriptions filter](#) [\(i\)](#)

[Search for any field...](#)

Subscription name ↑↓	Subscription ID ↑↓
Documentation Cross Sub Tests	
Documentation Testing 1	

2. In the left pane, select **Deployments**.

Microsoft Azure

Home > Documentation Testing 1

Subscription

Search (Ctrl+ /)

Billing

- Invoices

Settings

- Programmatic deployment
- Billing properties
- Resource groups
- Resources
- Preview features
- Usage + quotas
- Policies
- Management certificates
- My permissions
- Resource providers
- Deployments**
- Resource locks

Cancel subscription Rename Change directory Feedback

Essentials

Subscription ID : Microsoft (microsoft.onmicrosoft.com)

Directory : Microsoft (microsoft.onmicrosoft.com)

Status : Active

Parent management group :

Costs by resource [View details >](#)

Resource	Cost
dsqldbtest	\$0.44
arm	\$0.01
dscloudshellstg	\$0.01

3. Select one of the deployments from the deployment history.

Home > Documentation Testing 1

Documentation Testing 1 | Deployments

Subscription

Search (Ctrl+ /)

Billing

- Invoices

Settings

- Programmatic deployment
- Billing properties
- Resource groups
- Resources
- Preview features
- Usage + quotas
- Policies

Refresh | Cancel | Delete | View template

Filter by deployment name or resources in the deployment...

Deployment name	Status	Last modified
parent2	Succeeded	9/14/2021, 9:24:33 AM
testrg1	Succeeded	7/19/2021, 12:18:20 PM
newRG1	Succeeded	5/4/2021, 11:09:27 AM
subsub2	Succeeded	11/23/2020, 1:33:32 PM
subsub	Succeeded	11/23/2020, 1:26:11 PM
environment	Succeeded	4/27/2020, 3:56:41 PM
deployment	Succeeded	4/27/2020, 12:55:54 PM
tag1	Succeeded	2/14/2020, 6:38:08 AM

4. A summary of the deployment is displayed, including the correlation ID.

Home > Documentation Testing 1 >

newRG1 | Overview

Deployment

Search (Ctrl+ /) < Delete Cancel Refresh

Overview

Inputs Outputs Template

Your deployment is complete

Deployment name: newRG1 Subscription: Documentation Testing 1

Start time: 5/4/2021 11:09:24 AM Correlation ID: 53a4fa47-6f54-41fe-823c-e991b1c6db8c

Deployment details (Download)

Next steps

Go to resource group

Management group deployments

You can view the history of deployments to a management group.

Portal

1. Select the **management group** you want to examine. If you don't have sufficient permissions to view details about the management group, you won't be able to select it.

Microsoft Azure

Home >

Management groups

Search resources, services, and docs (G+/)

Search (Ctrl+ /) < Create Add subscription Refresh Expand / Collapse all Feedback

Overview Get started Settings

Use management groups to group subscriptions. Click on an existing group to drill in, view details and govern learn more.

Search by name or ID

Showing 2 subscriptions in 3 groups

Name	ID
> [] Tenant Root Group	

2. In the left pane, select **Deployments**.

Home > Management groups >

Tenant Root Group

Management group

Search (Ctrl+ /) <> + Create + Add subscription ⌂ Refresh ⌂ Rename ⌂

Overview

Subscriptions Name : Tenant Root Group
Resource Groups ID :
Resources Access Level : Owner
Activity Log Path : / Tenant Root Group
Access control (IAM)

Search by name or ID

Governance

Get started
Security
Policy
Deployments

Showing 2 subscriptions in 3 groups

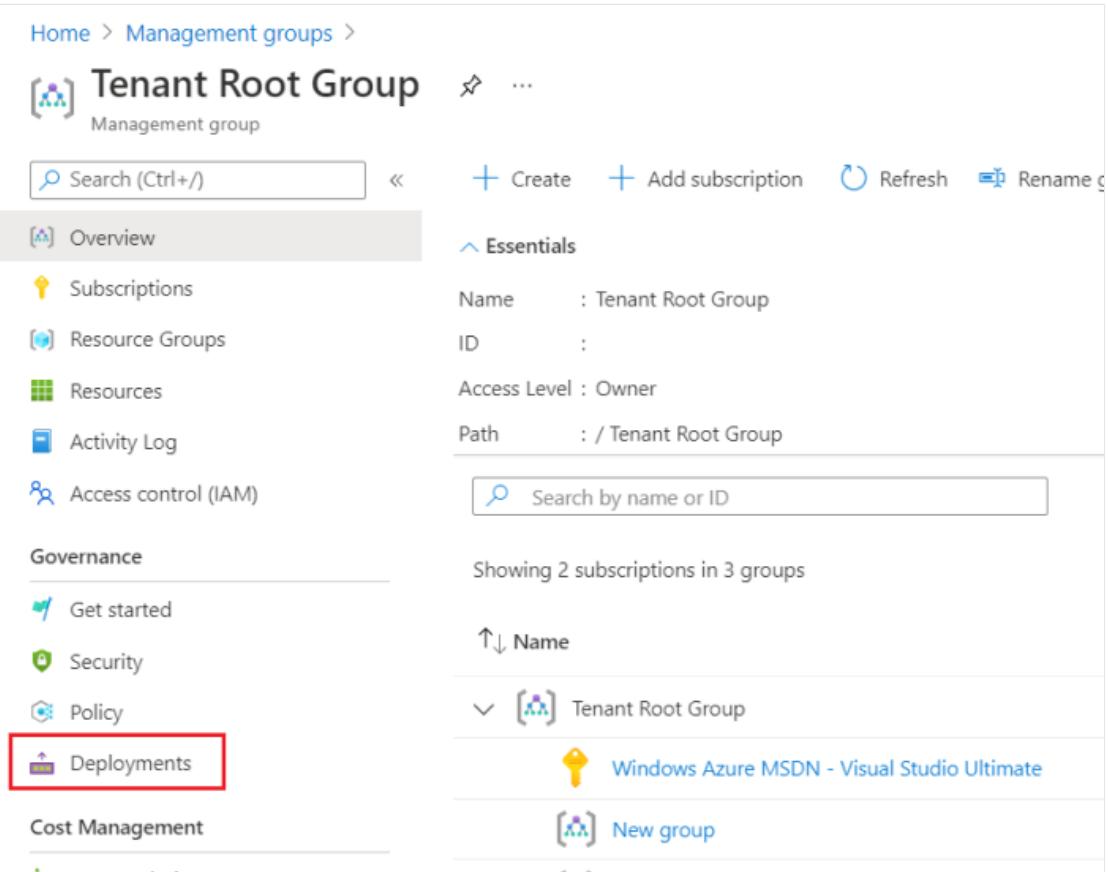
↑↓ Name

▼ [A] Tenant Root Group

[key] Windows Azure MSDN - Visual Studio Ultimate

[A] New group

Cost Management



3. Select one of the deployments from the deployment history.

Home > Management groups > Tenant Root Group

Tenant Root Group | Deployments

Management group

Search (Ctrl+ /) <> ⌂ Refresh | ⌂ Cancel | ⌂ Delete ⌂ View template

Filter by deployment name or resources in the deployment...

Deployment name	Status
<input type="checkbox"/> scope1	✓ Succeeded
<input type="checkbox"/> test2	! Failed (Error details)

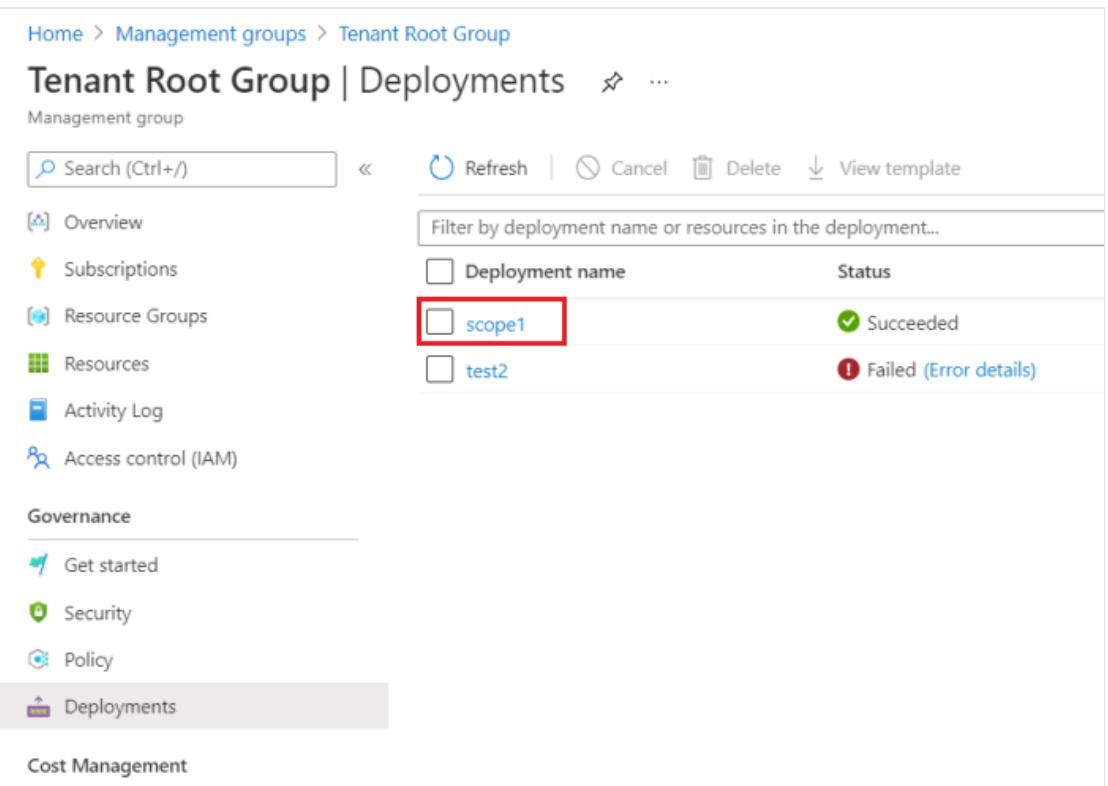
Overview

Subscriptions
Resource Groups
Resources
Activity Log
Access control (IAM)

Governance

Get started
Security
Policy
Deployments

Cost Management



4. A summary of the deployment is displayed, including the correlation ID.

The screenshot shows the Microsoft Azure Deployment Overview page. At the top, it displays the deployment name 'scope1' and a success message: 'Your deployment is complete'. Below this, it shows deployment details: 'Deployment name: scope1' and 'Tenant Example'. A red box highlights the 'Correlation ID: 11111111-1111-1111-1111-111111111111' field. A table below lists one resource: 'nested' (Type: Microsoft.Resources/deployments, Status: OK).

Tenant deployments

You can view the history of deployments to a tenant.

Portal

The portal doesn't currently show tenant deployments.

Deployment operations and error message

Each deployment can include multiple operations. To see more details about a deployment, view the deployment operations. When a deployment fails, the deployment operations include an error message.

Portal

1. On the summary for a deployment, select **Operation details**.

The screenshot shows the Microsoft Azure Deployment Overview page for 'ExampleDeployment'. It displays a failure message: 'Deployment failed. Click here for details →'. Below this, it shows deployment details: 'Deployment name: ExampleDeployment', 'Subscription: Documentation', and 'Resource group: ExampleGroup'. To the right, it shows the start time '11/9/2021, 2:29:19 PM' and correlation ID '11111111-1111-1111-1111-111111111111'. A red box highlights the 'Operation details' link under 'Deployment details (Download)'. A table below lists one operation: 'storage' (Type: Microsoft.Storage/storageAccounts, Status: BadRequest, Operation details: 'Operation details').

2. You see the details for that step of the deployment. When an error occurs, the details include the error message.

Operation details

Operation ID
/subscriptions/11111111-1111-1111-1111-111111111111/resourceGroups/Example...

Operation name
1234567890ABCDEF

Provisioning operation
Create

Provisioning state
Failed

Timestamp
11/9/2021, 2:29:20 PM

Duration
1 second

Tracking ID
11111111-1111-1111-1111-111111111111

serviceRequestId
11111111-1111-1111-1111-111111111111

Status
BadRequest

Status message

```
1  {
2      "status": "Failed",
3      "error": {
4          "code": "NoRegisteredProviderFound",
5          "message": "No registered resource provider found fo
6      }
7 }
```

Next steps

- For help resolve specific deployment errors, see [Troubleshoot common Azure deployment errors](#).
- To learn about how deployments are managed in the history, see [Automatic deletions from deployment history](#).

- To preview changes a template will make before you deploy, see [ARM template deployment what-if operation](#).

Use and manage extensions with the Azure CLI

Article • 06/05/2024

The Azure CLI offers the capability to load extensions. Extensions for the Azure CLI are characterized as Python wheels that aren't shipped as part of the CLI but run as CLI commands. With extensions, you gain access to experimental and prerelease commands along with the ability to write your own CLI interfaces. This article covers how to manage extensions and answers common questions about their use.

How to find extensions

To see the Azure CLI extensions provided and maintained by Microsoft, use the [az extension list-available](#) command.

```
Azure CLI
az extension list-available --output table
```

We also host a [list of extensions](#) on the documentation site.

How to install extensions

Install extensions manually

Once you have found an extension to install, use [az extension add](#) to get it. If the extension is listed in `az extension list-available`, you can install the extension by name.

```
Azure CLI
az extension add --name <extension-name>
```

If the extension is from an external resource or you have a direct link to it, provide the source URL or local path. The extension *must* be a compiled Python wheel file.

```
Azure CLI
```

```
az extension add --source <URL-or-path>
```

You can also build a private extension index following the format in [index.json](#), then set the extension index URL used by Azure CLI to it starting from version 2.20.0. After that, you can install the extension by name from the private extension index.

Azure CLI

```
az config set extension.index_url=<URL>
az extension add --name <extension-name>
```

Once an extension is installed, it's found under the value of the \$AZURE_EXTENSION_DIR shell variable. If this variable is unset, by default the value is \$HOME/.azure/cliextensions on Linux and macOS, and %USERPROFILE%\.azure\cliextensions on Windows.

Install extensions automatically

When you run an extension command that isn't installed, the Azure CLI can recognize the command you run, and automatically install the extension for you starting from version 2.10.0. This feature, referred to as **dynamic install**, is enabled by default since 2.12.0. You can also enable it through configuration for previous supported versions.

Azure CLI

```
az config set extension.use_dynamic_install=yes_prompt
```

Use the following configuration command to enable dynamic install without a prompt.

Azure CLI

```
az config set extension.use_dynamic_install=yes_without_prompt
```

Use the following configuration command to turn off the dynamic install feature to revert to the default behavior. The extension command returns a command-not-found error if the extension isn't installed.

Azure CLI

```
az config set extension.use_dynamic_install=no
```

By default, an extension command that prompts dynamic install will continue to run after the extension is installed. You can change the default behavior and make the command exit without a rerun by setting the `run_after_dynamic_install` property to `no`.

Azure CLI

```
az config set extension.run_after_dynamic_install=no
```

How to update extensions

If you install an extension by name, update it using [az extension update](#).

Azure CLI

```
az extension update --name <extension-name>
```

Otherwise, an extension can be updated from source by following the [Install extensions](#) instructions.

If you can't use CLI to resolve an extension name, uninstall it and attempt to reinstall. The extension could also have become part of the base CLI. Try updating the CLI as described in [Install the Azure CLI](#) and see if the extension's commands were added.

How to uninstall extensions

If you no longer need an extension, remove it with [az extension remove](#).

Azure CLI

```
az extension remove --name <extension-name>
```

You can also remove an extension manually by deleting it from the location where it was installed. The `$AZURE_EXTENSION_DIR` shell variable defines where modules are installed. If this variable is unset, by default the value is `$HOME/.azure/cliextensions` on Linux and macOS, and `%USERPROFILE%\.azure\cliextensions` on Windows.

Bash

```
rm -rf $AZURE_EXTENSION_DIR/<extension-name>
```

FAQ

Here are some answers to other common questions about CLI extensions.

What file formats are allowed for installation?

Currently, only compiled Python wheels can be installed as extensions.

Can extensions replace existing commands?

Yes. Extensions may replace existing commands, but before running a command that has been replaced the CLI issues a warning.

How can I tell if an extension is in prerelease?

An extension's documentation and versioning shows if it's in prerelease. Microsoft often releases preview commands as CLI extensions, with the option of moving them into the main CLI product later. When commands are moved out of extensions, the old extension should be uninstalled.

Can extensions depend upon each other?

No. Since the CLI doesn't guarantee a load order, dependencies might not be satisfied. Removing an extension doesn't affect any others.

Are extensions updated along with the CLI?

No. Extensions must be updated separately, as described in [Update extensions](#).

How to develop our own extension?

Refer to the official repository for more help. [Azure/azure-cli-extensions](#) ↗

Available Azure CLI extensions

Article • 10/08/2024

This article is a complete list of the available extensions for the Azure CLI which are supported by Microsoft. The list of extensions is also available from the CLI. To get it, run [az extension list-available](#):

```
Azure CLI
az extension list-available --output table
```

You will be prompted to install an extension on first use.

[Expand table](#)

Extension	Required Minimum CLI Version	Description	Status	Release Notes
acat	2.61.0	Microsoft Azure Command-Line Tools Acat Extension.	Preview	1.0.0b1
account	2.38.0	Microsoft Azure Command-Line Tools SubscriptionClient Extension	GA	0.2.5
acrquery	2.48.0	Microsoft Azure Command-Line Tools AcrQuery Extension	Preview	1.0.1
acrtransfer	2.0.67	Microsoft Azure Command-Line Tools Acrtransfer Extension	Preview	1.1.0
ad	2.15.0	Microsoft Azure Command-Line Tools DomainServicesResourceProvider Extension	GA	0.1.0
adp	2.40.0	Microsoft Azure Command-Line Tools Adp Extension.	Experimental	0.1.0
aem	2.19.1	Manage Azure Enhanced Monitoring Extensions for SAP	GA	0.3.0
ai-examples	2.2.0	Add AI powered examples to help content.	Preview	0.2.5
aks-preview	2.61.0	Provides a preview for upcoming AKS features	Preview	9.0.0b6
aksarc	2.32.0	Microsoft Azure Command-Line Tools HybridContainerService Extension	GA	1.2.23

Extension	Required Minimum CLI Version	Description	Status	Release Notes
akshybrid	2.32.0	Microsoft Azure Command-Line Tools HybridContainerService Extension	Experimental	0.1.2
alb	2.51.0	Microsoft Azure Command-Line Tools ALB Extension.	Preview	1.0.0
alertsmanagement	2.45.0	Microsoft Azure Command-Line Tools AlertsManagementClient Extension	Preview	1.0.0b1
alias	2.0.50.dev0	Support for command aliases	Preview	0.5.2
amg	2.61.0	Microsoft Azure Command-Line Tools Azure Managed Grafana Extension	GA	2.4.0
amlfs	2.49.0	Microsoft Azure Command-Line Tools Amlfs Extension.	GA	1.0.0
aosm	2.54.0	Microsoft Azure Command-Line Tools Aosm Extension	Preview	2.0.0b1
apic-extension	2.57.0	Microsoft Azure Command-Line Tools ApicExtension Extension.	GA	1.0.0
application-insights	2.55.0	Support for managing Application Insights components and querying metrics, events, and logs from such components.	GA	1.2.2
appservice-kube	2.34.1	Microsoft Azure Command-Line Tools App Service on Kubernetes Extension	Preview	0.1.10
arcappliance	2.51.0	Microsoft Azure Command-Line Tools Arcappliance Extension	GA	1.2.0
arCDATA	2.3.1	Tools for managing ArcData.	GA	1.5.18
astronomer	2.57.0	Microsoft Azure Command-Line Tools Astronomer Extension.	GA	1.0.0
attestation	2.55.0	Microsoft Azure Command-Line Tools AttestationManagementClient Extension	Preview	1.0.0
authV2	2.23.0	Microsoft Azure Command-Line Tools Authv2 Extension	GA	0.1.3
automanage	2.44.1	Microsoft Azure Command-Line Tools Automanage Extension.	Preview	0.1.2
automation	2.55.0	Microsoft Azure Command-Line Tools	Experimental	1.0.0b1

Extension	Required Minimum CLI Version	Description	Status	Release Notes
AutomationClient Extension				
azure-batch-cli-extensions	2.30.0	Additional commands for working with Azure Batch service	GA	7.0.1
azure-cli-ml	2.3.1	Microsoft Azure Command-Line Tools AzureML Command Module	GA	1.41.0
azure-devops	2.30.0	Tools for managing Azure DevOps.	GA	1.0.1
azure-firewall	2.61.0	Manage Azure Firewall resources.	GA	1.2.0
azure-iot	2.46.0	The Azure IoT extension for Azure CLI.	GA	0.25.0
azure-iot-ops	2.53.0	The Azure IoT Operations extension for Azure CLI.	Preview	0.7.0b1
azure-sphere	2.45.0	The Azure Sphere extension for Azure CLI.	GA	1.0.1
azurelargeinstance	2.57.0	Microsoft Azure Command-Line Tools Azurelargeinstance Extension.	Preview	1.0.0b4
azurestackhci	2.15.0	Microsoft Azure Command-Line Tools AzureStackHCI Extension	Experimental	0.2.9
baremetal-infrastructure	2.57.0	Microsoft Azure Command-Line Tools BaremetalInfrastructure Extension.	Preview	3.0.0b2
bastion	2.62.0	Microsoft Azure Command-Line Tools Bastion Extension.	GA	1.3.0
billing-benefits	2.43.0	Microsoft Azure Command-Line Tools BillingBenefits Extension.	Preview	0.1.0
blueprint	2.50.0	Microsoft Azure Command-Line Tools Blueprint Extension	Experimental	0.3.2
change-analysis	2.37.0	Microsoft Azure Command-Line Tools ChangeAnalysis Extension.	Preview	0.1.0
cli-translator	2.13.0	Translate ARM template to executable Azure CLI scripts.	Experimental	0.3.0
cloud-service	2.55.0	Microsoft Azure Command-Line Tools ComputeManagementClient Extension	GA	1.0.0
command-change	2.19.0	Microsoft Azure Command-Line Tools CommandChange Extension.	Preview	1.0.0b1

Extension	Required Minimum CLI Version	Description	Status	Release Notes
communication ↗	2.59.0	Microsoft Azure Command-Line Tools CommunicationServiceManagementClient Extension	GA	1.11.0 ↗
compute-diagnostic-rp ↗	2.56.0	Microsoft Azure Command-Line Tools ComputeDiagnosticRp Extension.	Preview	1.0.0b1 ↗
confcom ↗	2.26.2	Microsoft Azure Command-Line Tools Confidential Container Security Policy Generator Extension	GA	1.0.0 ↗
confidentialledger ↗	2.15.0	Microsoft Azure Command-Line Tools ConfidentialLedger Extension	GA	1.0.0 ↗
confluent ↗	2.56.0	Microsoft Azure Command-Line Tools ConfluentManagementClient Extension	GA	0.6.0 ↗
connectedk8s ↗	2.38.0	Microsoft Azure Command-Line Tools Connectedk8s Extension	GA	1.9.2 ↗
connectedmachine ↗	2.57.0	Microsoft Azure Command-Line Tools ConnectedMachine Extension	Preview	1.0.0b2 ↗
connectedvmware ↗	2.0.67	Microsoft Azure Command-Line Tools Connectedvmware Extension	GA	1.2.0 ↗
connection-monitor-preview ↗	2.0.80	Microsoft Azure Command-Line Connection Monitor V2 Extension	Preview	0.1.0 ↗
containerapp ↗	2.62.0	Microsoft Azure Command-Line Tools Containerapp Extension	Preview	1.0.0b3 ↗
cosmosdb-preview ↗	2.17.1	Microsoft Azure Command-Line Tools Cosmosdb-preview Extension	Preview	1.0.1 ↗
costmanagement ↗	2.55.0	Microsoft Azure Command-Line Tools CostManagementClient Extension	GA	1.0.0 ↗
csvmware ↗	2.0.67	Manage Azure VMware Solution by CloudSimple.	Preview	0.3.0 ↗
custom-providers ↗	2.3.1	Microsoft Azure Command-Line Tools Custom Providers Extension	Experimental	0.2.1 ↗
customlocation ↗	2.0.67	Microsoft Azure Command-Line Tools Customlocation Extension	GA	0.1.3 ↗
databox ↗	2.48.0	Microsoft Azure Command-Line Tools Databox Extension.	GA	1.1.0 ↗

Extension	Required Minimum CLI Version	Description	Status	Release Notes
databricks ↗	2.57.0	Microsoft Azure Command-Line Tools DatabricksClient Extension	GA	1.0.1 ↗
datadog ↗	2.17.0	Microsoft Azure Command-Line Tools MicrosoftDatadogClient Extension	Experimental	0.1.1 ↗
datafactory ↗	2.15.0	Microsoft Azure Command-Line Tools DataFactoryManagementClient Extension	GA	1.0.2 ↗
datamigration ↗	2.15.0	Microsoft Azure Command-Line Tools DataMigrationManagementClient Extension	Preview	1.0.0b2 ↗
dataprotection ↗	2.57.0	Microsoft Azure Command-Line Tools DataProtectionClient Extension	GA	1.5.3 ↗
datashare ↗	2.15.0	Microsoft Azure Command-Line Tools DataShareManagementClient Extension	Experimental	0.2.0 ↗
db-up ↗	2.0.46	Additional commands to simplify Azure Database workflows.	Preview	1.0.0b3 ↗
deploy-to-azure ↗	2.0.60	Deploy to Azure using Github Actions.	Preview	0.2.0 ↗
desktopvirtualization ↗	2.55.0	Microsoft Azure Command-Line Tools DesktopVirtualization Extension.	GA	1.0.0 ↗
dev-spaces ↗	2.1.0	Dev Spaces provides a rapid, iterative Kubernetes development experience for teams.	GA	1.0.6 ↗
devcenter ↗	2.51.0	Microsoft Azure Command-Line Tools DevCenter Extension	GA	6.0.1 ↗
diskpool ↗	2.15.0	Microsoft Azure Command-Line Tools StoragePoolManagement Extension	Experimental	0.2.0 ↗
dms-preview ↗	2.27.0	Support for new Database Migration Service scenarios.	Preview	0.15.0 ↗
dnc ↗	2.51.0	Microsoft Azure Command-Line Tools Dnc Extension.	Preview	0.2.1 ↗
dns-resolver ↗	2.39.0	Microsoft Azure Command-Line Tools DnsResolverManagementClient Extension	GA	0.2.0 ↗
durabletask ↗	2.61.0	Microsoft Azure Command-Line Tools Durabletask Extension.	Preview	1.0.0b1 ↗

Extension	Required Minimum CLI Version	Description	Status	Release Notes
dynatrace	2.41.0	Microsoft Azure Command-Line Tools Dynatrace Extension.	Preview	0.1.0
edgeorder	2.15.0	Microsoft Azure Command-Line Tools EdgeOrderManagementClient Extension	Experimental	0.1.0
edgezones	2.57.0	Microsoft Azure Command-Line Tools Edgezones Extension.	Preview	1.0.0b1
elastic	2.61.0	Microsoft Azure Command-Line Tools MicrosoftElastic Extension	Preview	1.0.0b3
elastic-san	2.61.0	Microsoft Azure Command-Line Tools ElasticSan Extension.	GA	1.1.0
eventgrid	2.51.0	Microsoft Azure Command-Line Tools EventGrid Command Module.	Preview	1.0.0b1
express-route-cross-connection	2.0.41	Manage customer ExpressRoute circuits using an ExpressRoute cross-connection.	GA	0.1.1
firmwareanalysis	2.55.0	Microsoft Azure Command-Line Tools Firmwareanalysis Extension.	Preview	1.0.0
fleet	2.61.0	Microsoft Azure Command-Line Tools Fleet Extension	GA	1.4.0
fluid-relay	2.39.0	Microsoft Azure Command-Line Tools FluidRelay Extension.	GA	0.1.0
footprint	2.11.0	Microsoft Azure Command-Line Tools FootprintMonitoringManagementClient Extension	Experimental	1.0.0
front-door	2.57.0	Manage networking Front Doors.	GA	1.2.0
functionapp	2.0.46	Additional commands for Azure Functions.	Preview	0.1.1
fzf	2.9.0	Microsoft Azure Command-Line Tools fzf Extension	Experimental	1.0.2
gallery-service-artifact	2.57.0	Microsoft Azure Command-Line Tools GalleryServiceArtifact Extension.	Preview	1.0.0b1
graphservices	2.49.0	Microsoft Azure Command-Line Tools Graphservices Extension.	Preview	1.0.0b1
guestconfig	2.3.1	Microsoft Azure Command-Line Tools	Experimental	0.1.1

Extension	Required Minimum CLI Version	Description	Status	Release Notes
		GuestConfigurationClient Extension		
hack	2.0.67	Microsoft Azure Command-Line Tools Hack Extension	Preview	0.4.3
hardware-security-modules	2.15.0	Microsoft Azure Command-Line Tools AzureDedicatedHSMResourceProvider Extension	Experimental	0.2.0
hdinsightonaks	2.57.0	Microsoft Azure Command-Line Tools Hdinsightonaks Extension.	Preview	1.0.0b3
healthbot	2.15.0	Microsoft Azure Command-Line Tools HealthbotClient Extension	Experimental	0.1.0
healthcareapis	2.15.0	Microsoft Azure Command-Line Tools HealthcareApisManagementClient Extension	GA	0.4.0
hpc-cache	2.3.0	Microsoft Azure Command-Line Tools StorageCache Extension	GA	0.1.5
hybridaks	2.32.0	Microsoft Azure Command-Line Tools HybridContainerService Extension	Experimental	0.2.4
image-copy-extension	2.0.24	Support for copying managed vm images between regions	GA	0.2.13
image-gallery	2.3.0	Support for Azure Image Gallery	Experimental	0.1.3
import-export	2.3.1	Microsoft Azure Command-Line Tools StorageImportExport Extension	Preview	1.0.0b1
informatica	2.61.0	Microsoft Azure Command-Line Tools Informatica Extension.	Preview	1.0.0b1
init	2.0.67	Microsoft Azure Command-Line Tools Init Extension	Preview	0.1.0
interactive	2.0.62	Microsoft Azure Command-Line Interactive Shell	Preview	0.5.3
internet-analyzer	2.0.67	Microsoft Azure Command-Line Tools Internet Analyzer Extension	Preview	1.0.0b1
ip-group	2.0.67	Microsoft Azure Command-Line Tools IpGroup Extension	Preview	0.1.2
k8s-configuration	2.15.0	Microsoft Azure Command-Line Tools K8s-configuration Extension	GA	2.0.0

Extension	Required Minimum CLI Version	Description	Status	Release Notes
k8s-extension	2.51.0	Microsoft Azure Command-Line Tools K8s-extension Extension	GA	1.6.1
k8s-runtime	2.57.0	Microsoft Azure Command-Line Tools K8sRuntime Extension.	GA	1.0.4
k8sconfiguration	2.15.0	Microsoft Azure Command-Line Tools K8sconfiguration Extension	Preview	0.2.4
kusto	2.15.0	Microsoft Azure Command-Line Tools KustoManagementClient Extension	GA	0.5.0
load	2.41.0	Microsoft Azure Command-Line Tools Load Testing Extension.	GA	1.0.1
log-analytics	2.61.0	Support for Azure Log Analytics query capabilities.	Preview	1.0.0b1
log-analytics-solution	2.50.0	Support for Azure Log Analytics Solution	GA	1.0.1
logic	2.55.0	Microsoft Azure Command-Line Tools Logic Extension.	GA	1.1.0
logz	2.15.0	Microsoft Azure Command-Line Tools MicrosoftLogz Extension	Experimental	0.1.0
maintenance	2.56.0	Microsoft Azure Command-Line Tools MaintenanceManagementClient Extension	Experimental	1.6.0
managedccfs	2.45.0	Microsoft Azure Command-Line Tools Managedccfs Extension.	Preview	0.2.0
managednetworkfabric	2.49.0	Support for managednetworkfabric commands based on 2023-06-15 API version.	GA	6.4.0
managementpartner	2.61.0	Support for Management Partner preview	GA	1.0.0
mdp	2.57.0	Microsoft Azure Command-Line Tools Mdp Extension.	Preview	1.0.0b2
mesh	2.50.0	Support for Microsoft Azure Service Fabric Mesh - Public Preview	Preview	1.0.0a1
microsoft-fabric	2.61.0	Microsoft Azure Command-Line Tools Microsoft Fabric Extension.	Preview	1.0.0b1
mixed-reality	2.49.0	Mixed Reality Azure CLI Extension.	Preview	0.0.5

Extension	Required Minimum CLI Version	Description	Status	Release Notes
ml	2.15.0	Microsoft Azure Command-Line Tools AzureMachineLearningWorkspaces Extension	GA	2.30.1
mobile-network	2.55.0	Microsoft Azure Command-Line Tools MobileNetwork Extension.	GA	1.0.0
monitor-control-service	2.61.0	Microsoft Azure Command-Line Tools MonitorClient Extension	GA	1.2.0
monitor-pipeline-group	2.61.0	Microsoft Azure Command-Line Tools MonitorPipelineGroup Extension.	Preview	1.0.0b1
multicloud-connector	2.61.0	Microsoft Azure Command-Line Tools MulticloudConnector Extension.	Preview	1.0.0b1
netappfiles-preview	2.61.0	Provides a preview for upcoming Azure NetApp Files (ANF) features.	Preview	1.0.0b3
network-analytics	2.51.0	Microsoft Azure Command-Line Tools NetworkAnalytics Extension.	Preview	1.0.0b1
networkcloud	2.61.0	Support for Azure Operator Nexus network cloud commands based on 2024-07-01 API version.	Preview	2.0.0b4
new-relic	2.57.0	Microsoft Azure Command-Line Tools NewRelic Extension.	GA	1.0.0
next	2.20.0	Microsoft Azure Command-Line Tools Next Extension	Experimental	0.1.3
nexusidentity	2.59.0	Microsoft Azure Command-Line Tools Nexusidentity Extension	Preview	1.0.0b1
nginx	2.64.0	Microsoft Azure Command-Line Tools Nginx Extension	Preview	2.0.0b6
notification-hub	2.49.0	Microsoft Azure Command-Line Tools Notification Hub Extension	GA	1.0.0a1
nsp	2.61.0	Microsoft Azure Command-Line Tools Nsp Extension.	Preview	1.0.0b2
offazure	2.15.0	Microsoft Azure Command-Line Tools AzureMigrateV2 Extension	Experimental	0.1.0
oracle-database	2.57.0	Microsoft Azure Command-Line Tools OracleDatabase Extension.	GA	1.0.0

Extension	Required Minimum CLI Version	Description	Status	Release Notes
orbital ↗	2.39.0	Microsoft Azure Command-Line Tools Orbital Extension.	GA	0.1.0 ↗
palo-alto-networks ↗	2.51.0	Microsoft Azure Command-Line Tools PaloAltoNetworks Extension.	Preview	1.1.1b1 ↗
partnercenter ↗	2.0.67	Microsoft Azure CLI Extension for Partner Center	Preview	0.2.4 ↗
peering ↗	2.3.1	Microsoft Azure Command-Line Tools PeeringManagementClient Extension	GA	1.0.0 ↗
portal ↗	2.3.1	Microsoft Azure Command-Line Tools Portal Extension	Experimental	0.1.3 ↗
powerbidedicated ↗	2.56.0	Microsoft Azure Command-Line Tools PowerBIDedicated Extension	Preview	1.0.0b1 ↗
providerhub ↗	2.57.0	Microsoft Azure Command-Line Tools ProviderHub Extension	Preview	1.0.0b1 ↗
purview ↗	2.15.0	Microsoft Azure Command-Line Tools PurviewManagementClient Extension	Preview	0.1.0 ↗
quantum ↗	2.41.0	Microsoft Azure Command-Line Tools Quantum Extension	Preview	1.0.0b3 ↗
qumulo ↗	2.48.0	Microsoft Azure Command-Line Tools Qumulo Extension.	GA	1.0.0 ↗
quota ↗	2.54.0	Microsoft Azure Command-Line Tools AzureQuotaExtensionAPI Extension	GA	1.0.0 ↗
rdbms-connect ↗	2.19.0	Support for testing connection to Azure Database for MySQL & PostgreSQL servers.	GA	1.0.6 ↗
redisenterprise ↗	2.61.0	Microsoft Azure Command-Line Tools RedisEnterprise Extension.	GA	1.2.0 ↗
reservation ↗	2.50.0	Microsoft Azure Command-Line Tools Reservation Extension	GA	0.3.1 ↗
resource-graph ↗	2.22.0	Support for querying Azure resources with Resource Graph.	GA	2.1.0 ↗
resource-mover ↗	2.50.0	Microsoft Azure Command-Line Tools ResourceMoverServiceAPI Extension	Preview	1.0.0b1 ↗

Extension	Required Minimum CLI Version	Description	Status	Release Notes
sap-hana ↗	2.0.46	Additional commands for working with SAP HanaOnAzure instances.	GA	0.6.5 ↗
scenario-guide ↗	2.20.0	Microsoft Azure Command-Line Tools Scenario Guidance Extension	Experimental	0.1.1 ↗
scheduled-query ↗	2.54.0	Microsoft Azure Command-Line Tools Scheduled_query Extension	Preview	1.0.0b1 ↗
scvmm ↗	2.15.0	Microsoft Azure Command-Line Tools SCVMM Extension	GA	1.1.1 ↗
self-help ↗	2.57.0	Microsoft Azure Command-Line Tools SelfHelp Extension.	Preview	0.4.0 ↗
sentinel ↗	2.37.0	Microsoft Azure Command-Line Tools Sentinel Extension.	Experimental	0.2.0 ↗
serial-console ↗	2.15.0	Microsoft Azure Command-Line Tools for Serial Console Extension	Preview	1.0.0b1 ↗
serviceconnector-passwordless ↗	2.60.0	Microsoft Azure Command-Line Tools Serviceconnector-passwordless Extension	GA	3.1.1 ↗
site-recovery ↗	2.51.0	Microsoft Azure Command-Line Tools SiteRecovery Extension.	Experimental	1.0.0 ↗
spring ↗	2.56.0	Microsoft Azure Command-Line Tools spring Extension	GA	1.25.1 ↗
spring-cloud ↗	2.45.0	Microsoft Azure Command-Line Tools spring-cloud Extension	GA	3.1.8 ↗
ssh ↗	2.45.0	SSH into Azure VMs using RBAC and AAD OpenSSH Certificates	GA	2.0.5 ↗
stack-hci ↗	2.54.0	Microsoft Azure Command-Line Tools AzureStackHCIClient Extension	GA	1.1.0 ↗
stack-hci-vm ↗	2.15.0	Microsoft Azure Command-Line Tools Stack-HCi-VM Extension	GA	1.3.0 ↗
standbypool ↗	2.57.0	Microsoft Azure Command-Line Tools Standbypool Extension.	GA	1.0.0 ↗
staticwebapp ↗	2.39.0	Microsoft Azure Command-Line Tools Staticwebapp Extension	Preview	1.0.0 ↗
storage-actions ↗	2.57.0	Microsoft Azure Command-Line Tools	Preview	1.0.0b1 ↗

Extension	Required Minimum CLI Version	Description	Status	Release Notes
StorageActions Extension.				
storage-blob-preview	2.27.0	Microsoft Azure Command-Line Tools Storageblobpreview Extension	Preview	1.0.0b1
storage-mover	2.57.0	Microsoft Azure Command-Line Tools StorageMover Extension.	GA	1.1.0
storage-preview	2.61.0	Provides a preview for upcoming storage features.	Preview	1.0.0b4
storagesync	2.55.0	Microsoft Azure Command-Line Tools MicrosoftStorageSync Extension	GA	1.0.1
stream-analytics	2.15.0	Microsoft Azure Command-Line Tools StreamAnalyticsManagementClient Extension	GA	1.0.0
subscription	2.0.30	Support for subscription management preview.	Preview	0.1.5
support	2.57.0	Microsoft Azure Command-Line Tools Support Extension	GA	2.0.0
terraform	2.61.0	Microsoft Azure Command-Line Tools Terraform Extension.	Preview	1.0.0b1
timeseriesinsights	2.50.0	Microsoft Azure Command-Line Tools TimeSeriesInsightsClient Extension	GA	1.0.0b1
traffic-collector	2.40.0	Microsoft Azure Command-Line Tools TrafficCollector Extension.	GA	0.1.3
trustedsigning	2.57.0	Microsoft Azure Command-Line Tools Trustedsigning Extension.	Preview	1.0.0b2
virtual-network-manager	2.61.0	Microsoft Azure Command-Line Tools NetworkManagementClient Extension	GA	1.3.0
virtual-network-tap	2.0.46	Manage virtual network taps (VTAP).	Preview	1.0.0b1
virtual-wan	2.55.0	Manage virtual WAN, hubs, VPN gateways and VPN sites.	GA	1.0.1
vm-repair	2.0.67	Auto repair commands to fix VMs.	GA	1.0.10
vmware	2.54.0	Azure VMware Solution commands.	GA	7.1.0
webapp	2.23.0	Additional commands for Azure AppService.	Preview	0.4.0

Extension	Required Minimum CLI Version	Description	Status	Release Notes
webpubsub	2.56.0	Microsoft Azure Command-Line Tools Webpubsub Extension	GA	1.6.0
workloads	2.57.0	Microsoft Azure Command-Line Tools Workloads Extension.	Preview	1.1.0b3

How to use the Azure CLI alias extension

Article • 06/05/2024

The alias extension allows users to define custom commands for the Azure CLI by using existing commands. Aliases help keep your workflow simple by allowing shortcuts. The Jinja2 template engine powers Azure CLI aliases and offers advanced argument processing.

ⓘ Note

The Alias Extension is in public preview. The features and configuration file format may change.

Install the alias extension

The minimum required Azure CLI version to use the alias extension is 2.0.28. To check your CLI version, run `az --version`. If you need to update your installation, follow the instructions in [Install the Azure CLI](#).

Install the alias extension with the [az extension add](#) command.

Azure CLI

```
az extension add --name alias
```

Verify the installation of the extension with [az extension list](#). If the alias extension was installed properly, it's listed in the command output.

Azure CLI

```
az extension list --output table --query '[].{Name:name}'
```

Output

Name

alias

Keep the alias extension up-to-date

The alias extension is under active development and new versions are released regularly. New versions aren't installed when you update the CLI. Install the updates for the extension with [az extension update](#).

```
Azure CLI
```

```
az extension update --name alias
```

Manage aliases for the Azure CLI

The alias extension lets you create and manage aliases for other CLI commands. To view all the available commands and parameter details, run the alias command with `--help`.

```
Azure CLI
```

```
az alias --help
```

Create simple alias commands

One use of aliases is for shortening existing command groups or command names. For example, you can shorten the `group` command group to `rg` and the `list` command to `ls`.

```
Azure CLI
```

```
az alias create --name rg --command group  
az alias create --name ls --command list
```

These newly defined aliases can now be used anywhere that their definition would be.

```
Azure CLI
```

```
az rg list  
az rg ls  
az vm ls
```

Don't include `az` as part of the alias command.

Aliases can also be shortcuts for complete commands. The next example lists available resource groups and their locations in table output:

```
Azure CLI
```

```
az alias create --name ls-groups --command "group list --query '[].  
{Name:name, Location:location}' --output table"
```

Now `ls-groups` can be run like any other CLI command.

```
Azure CLI
```

```
az ls-groups
```

Create an alias command with arguments

You can also add positional arguments to an alias command by including them as `{}
{{ arg_name }}` in the alias name. The whitespace inside the braces is required.

```
Azure CLI
```

```
az alias create --name "alias_name {{ arg1 }} {{ arg2 }} ..." --command  
"invoke_including_args"
```

The next example alias shows how to use positional arguments to get the public IP address for a VM.

```
Azure CLI
```

```
az alias create \  
  --name "get-vm-ip {{ resourceGroup }} {{ vmName }}" \  
  --command "vm list-ip-addresses --resource-group {{ resourceGroup }} --  
  name {{ vmName }} \  
  --query [0].virtualMachine.network.publicIpAddresses[0].ipAddress"
```

When running this command, you give values to the positional arguments.

```
Azure CLI
```

```
az get-vm-ip MyResourceGroup MyVM
```

You can also use environment variables in aliased commands, which are evaluated at runtime. The next example adds the `create-rg` alias, which creates a resource group in `eastus` and adds an `owner` tag. This tag is assigned the value of the local environment variable `USER`.

```
Azure CLI
```

```
az alias create \
    --name "create-rg {{ groupName }}" \
    --command "group create --name {{ groupName }} --location eastus --tags
owner=\$USER"
```

To register the environment variables inside the command of the alias, the dollar sign \$ must be escaped.

Process arguments using Jinja2 templates

Jinja2 [↗](#) performs the argument substitution in the alias extension. Jinja2 templates allow for manipulating the arguments.

With Jinja2 templates, you can write aliases that take different types of arguments than the underlying command. For example, you can make an alias that takes a storage URL. Then this URL is parsed to pass the account and container names to the storage command.

Azure CLI

```
az alias create \
    --name 'storage-ls {{ url }}' \
    --command "storage blob list
    --account-name {{ url.replace('https://', '').split('.')[0] }}
    --container-name {{ url.replace('https://', '').split('/')[1] }}"
```

To learn about the Jinja2 template engine, see [the Jinja2 documentation](#) [↗](#).

Alias configuration file

Another way to create and modify aliases is to alter the alias configuration file. Alias command definitions are written into a configuration file, located at

`$AZURE_CONFIG_DIR/alias`. The default value of `AZURE_CONFIG_DIR` is `$HOME/.azure` on macOS and Linux, and `%USERPROFILE%\.azure` on Windows. The alias configuration file is written in the INI configuration file format. The format for alias commands is:

ini

```
[alias_name]
command = invoked_commands
```

For aliases that have positional arguments, the format for alias commands is:

```
ini
```

```
[alias_name {{ arg1 }} {{ arg2 }} ...]  
command = invoked_commands_including_args
```

Create an alias command with arguments via the alias configuration file

The next example shows an alias for a command with arguments. This command gets the public IP address for a VM. Aliased commands must all be on a single line, and use all of the arguments in the alias name.

```
ini
```

```
[get-vm-ip {{ resourceGroup }} {{ vmName }}]  
command = vm list-ip-addresses --resource-group {{ resourceGroup }} --name  
{{ vmName }} --query  
[0].virtualMachine.network.publicIpAddresses[0].ipAddress
```

Uninstall the alias extension

To uninstall the extension, use the [az extension remove](#) command.

```
Azure CLI
```

```
az extension remove --name alias
```

If you uninstalled because a bug or other problem with the extension, [file a GitHub issue](#) so that we can provide a fix.

Begin your Azure CLI journey

Article • 03/20/2024

Welcome to the Azure CLI. We're glad you're here! You chose a tool that allows you to create, update, and delete Azure resources from a command-line. In this tutorial, you'll onboard with the Azure CLI and learn the following skills that help you to be successful in your Azure CLI journey:

- ✓ 1 - Get to know the Azure CLI (in this tutorial step)
 - Compare Azure CLI and PowerShell *tools*.
 - Review Azure CLI command components.
- ✓ 2 - [Prepare your environment for the Azure CLI](#)
 - Compare Bash and PowerShell *environments*.
 - Set environment variables.
 - Add a random ID to an Azure resource name.
- ✓ 3 - [Compare Azure CLI syntax differences in Bash, PowerShell and Cmd](#)
 - Understand differences in line continuation and variable usage.
 - Compare quoting differences and escape characters between environments.
 - Work with dates.
- ✓ 4 - [Populate variables for use in scripts](#)
 - Get (query) Azure resource properties.
 - Store the results of a command in a variable, such as a password or secret.
 - Store the contents of a JSON file in a variable, such as development or production settings.
- ✓ 5 - [Delete Azure resources at scale using a script](#)
 - Delete Azure resources using a FOR..EACH loop.

What is the Azure CLI?

The Azure Command-Line Interface (CLI) is a cross-platform command-line tool to connect to Azure and execute administrative commands on Azure resources. It allows the execution of commands through a terminal using interactive command-line prompts or a script.

For interactive use, first launch a shell such as Bash, PowerShell, or Cmd.exe on Windows, Linux or macOS. Then issue a command at the shell prompt. To automate

repetitive tasks, assemble the Azure CLI commands into a shell script using the script syntax of your chosen shell, and then execute the script.

What shell environment should I use?

Windows PowerShell, PowerShell, Cmd, and Bash are different shell environments. Your shell environment not only determines which tools you can use, but your shell also changes your command-line experience. For example, there are important quoting differences, line continuation characters, and variable syntax differences between shells.

The Azure CLI reference examples in this tutorial are written for Bash, PowerShell and Cmd to help you evaluate each shell experience. Here's a brief comparison:

[+] Expand table

Shell environment	Can I use Azure CLI?	Can I use Azure PowerShell?
CMD	Yes	
Bash	Yes	
Windows PowerShell	Yes	Yes
PowerShell	Yes	Yes

Azure Cloud Shell is a hosted shell environment that runs on an Ubuntu container. Cloud Shell provides two shell environments: Bash (with Azure CLI preinstalled) and PowerShell (with Azure PowerShell preinstalled).

How does the Azure CLI compare to Azure PowerShell?

Azure CLI and [Azure PowerShell](#) are both command-line tools that enable you to create and manage Azure resources, but they're distinctly different. Here's a brief comparison of the two tools:

[+] Expand table

Compare	Azure CLI	Azure PowerShell
What is it?	Cross-platform <i>command-line interface</i>	Cross-platform <i>PowerShell module</i>
Built on	Python	PowerShell

Compare	Azure CLI	Azure PowerShell
Install on	Windows, macOS, Linux	Windows, macOS, Linux
Run in	Bash shell, PowerShell 5 or 7, Windows Cmd.exe, Azure Cloud Shell, Docker container, and other Unix shells.	Windows PowerShell, PowerShell, Azure Cloud Shell, Docker container
Terms	Reference groups, reference subgroups and reference commands	Modules, cmdlets and aliases
Syntax example	az storage account create	New-AzStorageAccount
Learn more	What is Azure CLI?	What is Azure PowerShell?
Release notes	Azure CLI release notes	Azure PowerShell release notes

What are Azure CLI command components?

All Azure CLI reference commands are one of two reference types:

- **Core** components of the Azure CLI are part of the primary Azure CLI service and ship from Microsoft on a [release schedule](#).
- **Extensions** aren't shipped as part of the core Azure CLI but they run as CLI commands. With extensions, you gain access to experimental and prerelease commands along with the ability to write your own CLI interfaces. You're prompted to install an extension on first use.

The Azure CLI syntax has four parts. For example, this is how [az storage account local-user create](#) breaks down into syntax parts:

[\[+\] Expand table](#)

Term	Example
Reference group	az storage
Space-delimited reference subgroups	az storage account local-user
Reference command	az storage account local-user create
Reference command parameters	az storage account local-user create --name, --account-name, -resource-group

Get more details

Do you want more detail on one of the subjects covered in this tutorial step? Use the links in this table to learn more.

 Expand table

Subject	Learn more
Environments	Choose the right Azure command-line tool
Terms	Azure CLI terms and status
Extensions	Use and manage extensions with the Azure CLI

Next Step

Now that you've learned about shell environments and Azure CLI syntax, proceed to the next step to prepare your environment and execute your first Azure CLI reference command.

[Prepare your environment for the Azure CLI](#)

Prepare your environment for the Azure CLI

Article • 12/05/2023

In this tutorial step, you execute your first Azure CLI commands! This includes finding and setting your active subscription and setting default values. You also learn how to create resource groups containing a random ID to always guarantee a unique resource name.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

To complete this tutorial step, you need `contributor` or higher permissions on your subscription.

Install the Azure CLI

Get started by first selecting your environment.

- Use the Bash environment in [Azure Cloud Shell](#) by selecting the [Open Cloud Shell](#) button in the top right corner of each Azure CLI code block.
- If you prefer to run the Azure CLI reference commands locally, [install](#) the Azure CLI.

The current version of the Azure CLI is **2.55.0**. For information about the latest release, see the [release notes](#). To find your installed version and see if you need to update, run `az version`.

Sign in to Azure using the Azure CLI

There are several [authentication options](#) when working with the Azure CLI. The Azure CLI's default authentication method for logins uses a web browser and access token to sign in.

1. Run the `az login` command.

```
Azure CLI
az login
```

If the Azure CLI can open your default browser, it initiates [authorization code flow](#) and opens the default browser to load an Azure sign-in page.

Otherwise, it initiates the [device code flow](#) and instructs you to open a browser page at <https://aka.ms/devicelogin>. Then, enter the code displayed in your terminal.

If no web browser is available or the web browser fails to open, you may force device code flow with `az login --use-device-code`.

2. Sign in with your account credentials in the browser.

Find and change your active subscription

After logging into the Azure CLI, always check your current subscription. If you aren't working under the subscription you prefer, change your subscription using [az account set](#). Here's a code example to use:

```
Azure CLI

# see your current/default subscription
az account show

# find the list of subscriptions available to you
az account list --output table

# change your current/default subscription
az account set --subscription <mySubscriptionName>

# you can also set your subscription using a subscription ID
az account set --subscription <00000000-0000-0000-0000-000000000000>
```

Console output for `az account show` command:

```
Output

{
  "environmentName": "AzureCloud",
  "homeTenantId": "00000000-0000-0000-0000-000000000000",
  "id": "00000000-0000-0000-0000-000000000000",
  "isDefault": true,
  "managedByTenants": [],
  "name": "Your storage account name",
  "state": "Enabled",
  "tenantId": "00000000-0000-0000-0000-000000000000",
  "user": {
    "name": "yourName@contoso.com",
    "type": "user"
```

```
}
```

Learn about resource groups

A resource group is a container for Azure resources. To create a resource group, you need `contributor` permissions or higher in your subscription.

Create a resource group

1. A resource group name must be globally unique. First check if the name you want is available using the `az group exists` command. An output value of `false` means that the name is available.

Azure CLI

```
az group exists --name <myUniqueRGname>
```

2. Retrieve a list of supported regions for your subscription with the `az account list-locations` command. For a list of available Azure locations, see [Choose the right Azure region for you ↗](#).

Azure CLI

```
az account list-locations --query "[].{Region:name}" --output table
```

3. It's time to create your resource group! Use the `az group create` command.

Azure CLI

```
az group create --location <myLocation> --name <myUniqueRGname>
```

Create a resource group containing a random ID

When testing, it's best to create a resource group that contains a random ID in the name. Adding a random ID to your resource group name allows you to retest your code without having to wait for a prior resource group of the same name to be removed from Azure.

Bash and PowerShell variable syntax is different. Copy the correct script for your environment.

Bash

Azure CLI

```
let "randomIdentifier=$RANDOM*$RANDOM"
location="eastus"
resourceGroup="msdocs-tutorial-rg-$randomIdentifier"
az group create --name $resourceGroup --location $location --output json
```

Bash and PowerShell console output:

Output

```
{
  "id": "/subscriptions/00000000-0000-0000-0000-
0000000000/resourceGroups/msdocs-tutorial-rg-00000000",
  "location": "eastus",
  "managedBy": null,
  "name": "msdocs-tutorial-rg-00000000",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

Set environment variables

The Azure CLI offers several options to allow you to reuse common parameter values. These default values are stored in environment variables and are unique for each sign in.

1. Set your default resource group.

Azure CLI

```
az config set defaults.group=<msdocs-tutorial-rg-0000000>
```

2. Set multiple environment variables at once.

Many Azure CLI parameters take multiple values separated by a space. Configuration values are one such instance. The next example sets both the

.location and .group defaults that are used by the --location and --resource-group parameters of every Azure CLI command.

Azure CLI

```
az config set defaults.location=westus2 defaults.group=<msdocs-tutorial-rg-0000000>
```

3. Set your default output.

When you chose to work in Azure Cloud Shell, or install the Azure CLI locally, the default output is automatically set to json. However, this is one of the most important defaults to understand and set. **Output determines what appears on your console and what is written to your log file.** Always use an output of none when you're creating resources that return keys, passwords and secrets.

Azure CLI

```
az config set core.output=none
```

In this tutorial, we aren't working with secrets. Set the default back to json so you can see the returned output of each reference command in this tutorial.

Azure CLI

```
az config set core.output=json
```

4. Learn to use az init.

The Azure CLI has a reference command that walks you through configuring your environment. Type az init in your console and press Enter. Follow the prompts provided.

Azure CLI

```
az init
```

The first nice thing about az init is that it gives you all of your current settings! Here's example output:

Output

```
Your current config settings:
```

```
Output format: JSON  
[core.output = json]
```

```
Standard error stream (stderr): All events  
[core.only_show_errors = false]
```

```
Error output: Show recommendations  
[core.error_recommendation = on]
```

```
Syntax highlighting: On  
[core.no_color = false]
```

```
Progress Bar: On  
[core.disable_progress_bar = false]
```

```
Select an option by typing it's number
```

```
[1] Optimize for interaction
```

```
    These settings improve the output legibility and optimize for  
    human interaction
```

```
[2] Optimize for automation
```

```
    These settings optimize for machine efficiency
```

```
[3] Customize settings
```

```
    A walk through to customize common configurations
```

```
[4] Exit (default)
```

```
    Return to the command prompt
```

```
? Your selection:
```

5. Find and read your configuration file.

If you work under a "trust but verify" mindset, you want to know where your configuration files are stored and what they contain. The configuration file itself is located at `$AZURE_CONFIG_DIR/config`. The default value of `AZURE_CONFIG_DIR` is `$HOME/.azure` on Linux and macOS, and `%USERPROFILE%\.azure` on Windows. Find your config file now and see what it contains.

Get more details

Do you want more detail on one of the subjects covered in this tutorial step? Use the links in this table to learn more.

Subject	Learn more
Environments	Choose the right Azure command-line tool
Sign in options	Sign in with Azure CLI
Terms	Azure CLI terminology and support levels
Subscriptions	Manage subscriptions using the Azure CLI
Resource groups	Manage resource groups using the Azure CLI
Configurations	Configure the Azure CLI
Azure roles	Azure roles, Microsoft Entra roles, and classic subscription administrator roles

Next Step

Now that you've learned how to configure your environment, proceed to the next step to learn the scripting differences between Bash, PowerShell and Cmd.

[Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#)

Learn Azure CLI syntax differences in Bash, PowerShell and Cmd

Article • 06/27/2024

Azure CLI commands can be executed in both [Bash](#), [PowerShell](#), and Windows command shell ([Cmd](#)) scripting languages. However, there are subtle scripting differences. In this tutorial step, learn how to create your first Azure Storage Account and format parameter values for all three scripting languages.

Prerequisites

- You completed the prerequisites to [prepare your environment](#).
- You have access to a resource group with `contributor` or higher permissions at a resource group level.

Be aware of line continuation characters

Most Azure CLI documentation is written and tested in Bash using Azure Cloud Shell. One of the first things to remember when copying Azure CLI syntax is to verify the line continuation characters for your chosen scripting language as they aren't interchangeable.

[+] [Expand table](#)

scripting language	Line continuation character
Bash	Backslash (\)
PowerShell	Backtick (`)
Cmd	Carrot (^)

💡 Tip

The **Copy** button in the upper right corner of Azure CLI code blocks removes the backslash (\) and the backtick (`) by design. If you want to copy a formatted code block, use your keyboard or mouse to select and copy the example.

Understand syntax differences when using variables

The syntax for using variables varies slightly between scripting languages. Here's a comparison:

[+] Expand table

Use case	Bash	PowerShell	Cmd
Create variable	variableName=varValue	\$variableName="varValue"	set variableName=varValue
Use variable as parameter value	variableName	\$variableName	%variableName%
Use variable in parameter	'\$variableName' --query	'\$variableName'	'\$variableName'

There are several different ways to return variable information to your console screen, but `echo` works in most circumstances. Here's a comparison:

- **Bash:** echo \$varResourceGroup
- **PowerShell:** echo \$varResourceGroup
- **Cmd:** echo %varResourceGroup%

In step three, [Populate variables for use in scripts](#), you work through in-depth examples of variable syntax.

Learn about quoting differences between scripting languages

Every Azure CLI parameter is a string. However, each scripting language has its own rules for handling single and double quotes, spaces and parameter values.

[+] Expand table

String value	Azure CLI	PowerShell	Cmd
Text	'text' or "text"	'text' or "text"	"text"

String value	Azure CLI	PowerShell	Cmd
Number	\`50\`	``50``	`50`
Boolean	\`true\`	``false``	'true'
Date	'2021-11-15'	'2021-11-15'	'2021-11-15'
JSON	'{"key": "value"}' or " {"key": "value"}"	'{"key": "value"}' or '{"key": "value"}' or '{"key": "value"}'	" {"key": "value"}"

Many Azure CLI parameters accept a space-separated list of values. This impacts quoting.

- **Unquoted space-separated list:** --parameterName firstValue secondValue
- **Quoted space-separated list:** --parameterName "firstValue" "secondValue"
- **Values that contain a space:** --parameterName "value1a value1b" "value2a value2b" "value3"

If you aren't sure how your string will be evaluated by your scripting language, return the value of a string to your console or use `--debug` as explained in [Debug Azure CLI reference commands](#).

Create a storage account to apply what you've learned

The remainder of this tutorial step demonstrates quoting rules in Azure CLI commands, and uses the resource group created in [Prepare your environment for the Azure CLI](#). Substitute `<msdocs-tutorial-rg-00000000>` with the name of your resource group.

Create an Azure storage account to use in this tutorial. This example assigns a random ID to the storage account name, but if you want to use a different name, see [Storage account overview](#) for storage account name rules.

This next script example demonstrates scripting language-specific syntax for the following:

- Line continuation
- Variable usage
- Random identifiers
- `echo` command

Bash

Azure CLI

```
# Variable block
let "randomIdentifier=$RANDOM*$RANDOM"
location="eastus"
resourceGroup="<msdocs-tutorial-rg-00000000>"
storageAccount="msdocssa$randomIdentifier"

# Create a storage account.
echo "Creating storage account $storageAccount in resource group
$resourceGroup"
az storage account create --name $storageAccount \
    --resource-group $resourceGroup \
    --location $location \
    --sku Standard_RAGRS \
    --kind StorageV2 \
    --output json
```

① Note

Did you just receive a "Subscription not found" error? This error occurs when `Microsoft.Storage` is not registered in the active subscription. To register a resource provider, see [Azure resource providers and types](#).

The Azure CLI returns over 100 lines of JSON as output when a new storage account is created. The following JSON dictionary output has fields omitted for brevity.

Output

```
{
  "accessTier": "Hot",
  "allowBlobPublicAccess": false,
  "creationTime": "yyyy-mm-ddT19:14:26.962501+00:00",
  "enableHttpsTrafficOnly": true,
  "id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/
msdocs-tutorial-rg-
00000000/providers/Microsoft.Storage/storageAccounts/msdocssa00000000",
  "keyCreationTime": {
    "key1": "yyyy-mm-ddT19:14:27.103127+00:00",
    "key2": "yyyy-mm-ddT19:14:27.103127+00:00"
  },
  "kind": "StorageV2",
  "location": "eastus",
  "name": "msdocssa00000000",
  "primaryEndpoints": {
```

```
    "blob": "https://msdocssa00000000.blob.core.windows.net/",
},
"primaryLocation": "eastus",
"provisioningState": "Succeeded",
"resourceGroup": "msdocs-tutorial-rg-00000000",
"sku": {
    "name": "Standard_RAGRS",
    "tier": "Standard"
},
"statusOfPrimary": "available",
"statusOfSecondary": "available",
"tags": {},
"type": "Microsoft.Storage/storageAccounts"
}
```

Create tags to practice quoting differences

Using [az storage account update](#), add tags to help you identify your storage account and learn about quoting differences. These script examples demonstrate scripting language-specific syntax for the following:

- Values containing spaces
- Quoting blank spaces
- Escaping special characters
- Using variables

The `--tags` parameter accepts a space-separated list of key:value pairs. Substitute `<msdocs-tutorial-rg-00000000>` with the name of your resource group and `<msdocssa00000000>` with the name of your Azure storage account.

Bash

Azure CLI

```
# Create new tags. This syntax works with or without quotes around each
# key-value pair.
az storage account update --name <msdocssa00000000> \
    --resource-group <msdocs-tutorial-rg-00000000>
\ \
    --tags Team=t1 Environment=e1

# Create new tags containing spaces. You must use quotes.
az storage account update --name <msdocssa00000000> \
    --resource-group <msdocs-tutorial-rg-00000000>
\ \
    --tags "Floor number=f1" "Cost center=cc1"
```

```

# Create a new tag with an empty value.
az storage account update --name <msdocssa0000000> \
    --resource-group <msdocs-tutorial-rg-00000000>
\ \
    --tags "Department='''"

# Create a new tag containing special characters resulting in "Path":
"$G:\\myPath".
az storage account update --name <msdocssa0000000> \
    --resource-group <msdocs-tutorial-rg-00000000>
\ \
    --tags "Path=$G:\\myPath"

# Create a tag from a variable.
newTag="tag1=tag value with spaces"
az storage account update --name <msdocssa0000000> \
    --resource-group <msdocs-tutorial-rg-00000000>
\ \
    --tags "$newTag"

```

If you don't want to overwrite previous tags while you work through this tutorial step, use the `az tag update` command setting the `--operation` parameter to `merge`.

Azure CLI

```

# Get the resource ID of your storage account.
saID=$(az resource show --resource-group <msdocs-tutorial-rg-00000000> \
    --name <msdocssa0000000> \
    --resource-type
Microsoft.Storage/storageAccounts \
    --query "id" \
    --output tsv)

echo My storage account ID is $saID

# Append new tags.
az tag update --resource-id $saID \
    --operation merge \
    --tags <>tagName>=<tagValue>

# Get a list of all tags.
az tag list --resource-id $saID

```

Compare more scripting language-specific scripts

Take a deeper look at these script differences. These examples demonstrate quoting differences for the following:

- Pass a JSON string as a parameter value
- Filter results with the `--query` parameter
 - Numbers
 - Boolean values
 - Dates

Bash

Example of a parameter containing a JSON string. This script is given for future reference as we are not working with `az rest` in this tutorial.

Azure CLI

```
az rest --method patch \
    --url
https://management.azure.com/subscriptions/<mySubscriptionID>/resourceGroups/<myResourceGroup>/providers/Microsoft.HybridCompute/machines/<machineName>?api-version=yyyy-mm-dd-preview \
    --resource https://management.azure.com/ \
    --headers Content-Type=application/json \
    --body '{"properties": {"agentUpgrade": {"enableAutomaticUpgrade": false}}}'
```

Example of filtering for a numeric value. Unless you have a VM in your current subscription, this example is given for future reference.

Azure CLI

```
az vm list --resource-group <myResourceGroup> \
    --query "[?storageProfile.osDisk.diskSizeGb >=\`50\`].
{Name:name, admin:osProfile.adminUsername,
DiskSize:storageProfile.osDisk.diskSizeGb}" \
    --output table
```

Example of filtering a boolean value using the storage account created in this tutorial.

Azure CLI

```
az storage account list --resource-group <msdocs-tutorial-rg-00000000> \
    --query "[?allowBlobPublicAccess == \`true\`].id"
```

Examples of filtering a date using the storage account created in this tutorial.

Azure CLI

```
# include time
az storage account list --resource-group <msdocs-tutorial-rg-00000000> \
    --query "[?creationTime >='2021-11-15T19:14:27.103127+00:00'].{saName:name, saID: id, sku: sku.name}"

# exclude time
az storage account list --resource-group <msdocs-tutorial-rg-00000000> \
    --query "[?creationTime >='2021-11-15'].{saName:name, saID: id, sku: sku.name}"

# subtract days and use a variable
saDate=$(date +%F -d "-30days")
az storage account list --resource-group <msdocs-tutorial-rg-00000000> \
    --query "[?creationTime >='$saDate'].{saName:name, saID: id, sku: sku.name}"
```

Debug Azure CLI reference commands

Use `--debug` parameter

The Azure CLI offers a `--debug` parameter that can be used with any command. Debug output is extensive, but it gives you more information on execution errors. Use the Bash `clear` command to remove console output between tests.

These examples reveal the actual arguments received by the Azure CLI in Python syntax.

Bash

This example is **correct** in both Bash and PowerShell.

Azure CLI

```
az '{"key":"value"}' --debug
```

See what the Azure CLI is interpreting in the `Command arguments` line of the output.

Output

```
Command arguments: ['{"key":"value"}', '--debug']
```

This second example is also **correct**. Use the Bash `clear` command to remove console output between tests.

Azure CLI

```
clear  
az "{\"key\":\"value\"}" --debug
```

Output

```
Command arguments: ['{"key": "value"}', '--debug']
```

These next two examples are **incorrect** as quotes and spaces are interpreted by Bash.

[Expand table](#)

Incorrect format	Problem	Console output
az {"key": "value"} --debug	Missing single quotes or escape characters	Command arguments: ['{key:value}', '--debug']
az {"key": "value"} --debug	Missing single quotes or escape characters, and contains extra space	Command arguments: ['{key:', 'value}', '--debug']

Use `echo` command

Although `--debug` tells you exactly what the Azure CLI is interpreting, a second option is to return the value of an expression to your console. This method is helpful when verifying the results of `--query` that is covered in detail in [Populate variables for use in scripts](#).

Bash

Azure CLI

```
strExpression='{"key": "value"}'  
echo $strExpression
```

Output

```
{"key": "value"}
```

Troubleshooting

Here are common errors when an Azure CLI reference command syntax isn't written properly:

- "Bad request ...{something} is invalid" might be caused by a space, single or double quotation mark, or lack of a quote.
- "Unexpected token..." is seen when there's an extra space or quote.
- "Invalid jmespath_type value" error often comes from incorrect quoting in the `--query` parameter.
- "Variable reference is not valid" is received when a string isn't formatted properly often due to concatenation or a missing escape character.
- "Unrecognized arguments" is often caused by an incorrect line continuation character.
- "Missing expression after unary operator" is seen when a line continuation character is missing.

Get more details

Do you want more detail on one of the subjects covered in this tutorial step? Use the links in this table to learn more.

[] Expand table

Subject	Learn more
Scripting differences	Quoting differences between scripting languages
	Bash quoting rules ↗
	PowerShell quoting rules
	Considerations for running the Azure CLI in a PowerShell scripting language
	Windows command-line tips ↗
Parameters	Use quotation marks in Azure CLI parameters
	Find more syntax examples of Bash, PowerShell and Cmd in Query command output using JMESPath

Next Step

Now that you learned how to write Azure CLI syntax for Bash, PowerShell and Cmd, proceed to the next step to learn how to extract values to a variable.

[Populate variables for use in scripts](#)

 Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Populate variables for use in scripts

Article • 12/05/2023

In this tutorial step, learn to work with variables:

- Execute an Azure CLI command and store output in a variable.
- Read a local JSON file and store property values in a variable.

Some common use-cases for variables are:

- Get information about an existing Azure resource, such as a resource ID.
- Get output from an Azure CLI command, such as a password.
- Get JSON objects for environment information, such as development, stage and production IDs.

The variable can then be used in Azure CLI to perform build and destroy jobs at scale.

Prerequisites

- You have access to a resource group and storage account with `reader` or higher permissions at a storage account level.

Get command output using JMESPath query

Get information about an existing Azure resource using the `--query` parameter of the `show` command. A [JMESPath query](#) is executed and one or more property values of an Azure resource are returned.

Tip

The syntax for `--query` is case sensitive *and environment-specific*. If you receive empty results, check your capitalization. Avoid quoting errors by applying the rules you learned in [Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#)

Unless the `--output` parameter is specified, these examples rely on a default output configuration of `json` set in [Prepare your environment for the Azure CLI](#)

Get JSON dictionary properties of an Azure resource

Using the storage account created in [Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#), get the `primaryEndpoints` of your new storage account.

Azure CLI

```
az storage account show --resource-group <msdocs-tutorial-rg-00000000> \
--name <msdocssa00000000> \
--query primaryEndpoints
```

Console JSON dictionary output:

Output

```
{  
  "blob": "https://msdocssa00000000.blob.core.windows.net/",  
  "dfs": "https://msdocssa00000000.dfs.core.windows.net/",  
  "file": "https://msdocssa00000000.file.core.windows.net/",  
  "internetEndpoints": null,  
  "microsoftEndpoints": null,  
  "queue": "https://msdocssa00000000.queue.core.windows.net/",  
  "table": "https://msdocssa00000000.table.core.windows.net/",  
  "web": "https://msdocssa00000000.z13.web.core.windows.net/"  
}
```

Get individual JSON objects

Specify a comma-delimited list of storage account properties to return individual properties in an array (list).

Azure CLI

```
az storage account show --resource-group <msdocs-tutorial-rg-00000000> \
--name <msdocssa00000000> \
--query "[id, primaryLocation,
primaryEndpoints.blob, encryption.services.blob.lastEnabledTime]"
```

Console JSON array output:

Output

```
[  
  "/subscriptions/00000000-0000-0000-0000-  
000000000000/resourceGroups/msdocs-tutorial-rg-  
00000000/providers/Microsoft.Storage/storageAccounts/ msdocssa00000000",  
  "eastus",  
  "https://msdocssa00000000.blob.core.windows.net/",
```

```
"yyyy-mm-ddT19:11:56.399484+00:00"
```

```
]
```

Rename properties

Rename properties using curly brackets (`{}`) and a comma-delimited list. The new property names can't contain spaces. This example returns output in `table` format.

Azure CLI

```
az storage account show --resource-group <msdocs-tutorial-rg-00000000> \
    --name <msdocssa00000000> \
    --query "{saName:name, saKind:kind,
    saMinTLSVersion:minimumTlsVersion}" \
    --output table
```

Console table output. The first letter of each column is capitalized in `--output table`:

Output

SaName	SaKind	SaMinTLSversion
msdocssa0000000000	StorageV2	TLS1_0

Filter query results

Combine what you learned about quoting with what you just learned about `--query`. These examples apply a filter.

Bash

In Bash, you can't have a space before or after the equal (=) sign. You can opt to use quotes around the variable value, so `msdocs-tutorial-rg-00000000` and `"msdocs-tutorial-rg-00000000"` are both correct.

Azure CLI

```
rgName=<msdocs-tutorial-rg-00000000>

# Get a list of all Azure storage accounts that allow blob public
# access.
# Notice the backticks and escape characters needed for boolean values.
az storage account list --resource-group $rgName \
    --query "[?allowBlobPublicAccess ==
```

```
\`true\`].name"

# Get a list of Azure storage accounts that were created in the last 30
days. Return the results as a table.
saDate=$(date +%F -d "-30days")
az storage account list --resource-group $rgName \
    --query "[?creationTime >='$saDate'].
{saName:name, createdTimeStamp:creationTime}" \
    --output table

# Get a list of Azure storage accounts created in this tutorial
az storage account list --resource-group $rgName \
    --query "[?contains(name, 'msdocs')].
{saName:name, saKind:kind, saPrimaryLocation:primaryLocation,
createdTimeStamp:creationTime}" \
    --output table
```

Create a new Azure resource storing output in a variable

Learning to store command output in a variable is beneficial when creating Azure resources that output secrets that should be protected. For example, when you create a service principal, reset a credential, or get an Azure key vault secret, the command output should be protected.

Create a new Azure Key Vault and secret returning command output to a variable. Your Azure Key Vault name must be globally unique, so the `$RANDOM` identifier is used in this example. For more Azure Key Vault naming rules, see [Common error codes for Azure Key Vault](#).

These examples use `echo` to verify variable values because this is a teaching tutorial. Don't use `echo` for secret and password values in production-level environments.

Bash

Azure CLI

```
# Set your variables.
let "randomIdentifier=$RANDOM*$RANDOM"
rgName=<msdocs-tutorial-rg-00000000>
kvName=msdocs-kv-$randomIdentifier
location=eastus

# Set your default output to none
az config set core.output=None
```

```

# Create a new Azure Key Vault returning the Key Vault ID
myNewKeyVaultID=$(az keyvault create --name $kvName --resource-group
$rgName --location $location --query id --output tsv)
echo "My new Azure Key Vault ID is $myNewKeyVaultID"

# Wait about 1 minute for your Key Vault creation to complete.

# Create a new secret returning the secret ID
kvSecretName=<myKVSecretName>
kvSecretValue=<myKVSecretValue>
myNewSecretID=$(az keyvault secret set --vault-name $kvName --name
$kvSecretName --value $kvSecretValue --query id --output tsv)
echo "My new secret ID is $myNewSecretID"

# Reset your default output to json
az config set core.output=json

```

Get the contents of a JSON file and store it in a variable

This next section is a "stretch task" for an onboarding tutorial. However, to manage Azure resources in development, stage and production environments, you often need to read a configuration file.

Are you ready to stretch your Azure CLI skills? Create a JSON file containing the following JSON, or your file contents of choice. Save the text file to your local drive. If you're working in Azure Cloud Shell, use the `upload/download files` icon in the menu bar to store the text file in your cloud storage drive.

JSON

```
{
  "environments": {
    "dev": [
      {
        "id": "1",
        "kv-secretName": "dev1SecretName",
        "status": "inactive",
      },
      {
        "id": "2",
        "kv-secretName": "dev2SecretName",
        "status": "active"
      }
    ],
    "stg": {
      "id": "3",
      "kv-secretName": "dev3SecretName"
    }
  }
}
```

```
        },
      "prod": {
        "id": "4",
        "kv-secretName": "dev4SecretName"
      }
    }
}
```

Store the contents of your JSON file in a variable for further use in your Azure CLI commands. In this example, change `msdocs-tutorial.json` to the name of your file.

Don't save the `echo` command in production-level scripts as the output is saved in your log file.

Bash

This Bash script was tested in [Azure Cloud Shell](#) and depends on the Bash [jq](#) ↗ which must be installed in your environment.

Bash

```
# Show the contents of a file in the console
fileName=msdocs-tutorial.json
cat $fileName | jq

# Get a JSON dictionary object
stgKV=$(jq -r '.environments.stg."kv-secretName"' $fileName)
echo $stgKV

# Filter a JSON array
devKV=$(jq -r '.environments.dev[] | select(.status=="active") | ."kv-
secretName"' $fileName)
echo $devKV
```

Did you just receive a "jq command not found" error? This is because this script depends on the Bash [jq](#) ↗ command. [Install jq](#) ↗ in your environment, or run this script in Azure Cloud Shell.

You now have an environment-specific Azure Key Vault secret name stored in a variable, and you can use it to connect to Azure resources. This same method is good for IP addresses of Azure VMs and SQL Server connection strings when you want to reuse Azure CLI scripts.

Get more details

Do you want more detail on one of the subjects covered in this tutorial step? Use the links in this table to learn more.

[] [Expand table](#)

Subject	Learn more
Variables	See advanced examples in Use the Azure CLI successfully - Pass values to another command
	Read a good overview of variables in How to use variables in Azure CLI commands .
Querying	Find a wide range of examples in How to query Azure CLI command output using a JMESPath query .
	Take a deeper dive in Bash using <code>--query</code> in Learn to use Bash with the Azure CLI .
Azure key vault	About Azure Key Vault
	Provide access to Key Vault keys, certificates, and secrets with an Azure role-based access control.
	Common error codes for Azure Key Vault
PowerShell	Reference links: Get-content , Where-Object , Select-Object

Next Step

Now that you understand how to use variables to store Azure CLI command output and JSON property values, proceed to the next step to learn how to use scripts to delete Azure resources.

[Delete Azure resources at scale using a script](#)

Delete Azure resources at scale using a script

Article • 12/05/2023

In this tutorial step, learn to delete multiple Azure resources using a Bash or PowerShell script. This skill is especially helpful when you're managing a large number of Azure resources and need to tear down development or testing environments.

Prerequisites

- You created at least two storage accounts as described in [Learn Azure CLI syntax differences in Bash, PowerShell and Cmd](#).

Delete a resource group by name

Using random IDs and running these tutorial steps creates test resource groups that can be removed. The easiest way to clean up Azure resources is to delete the resource group. However, when you delete a resource group, *you delete every object inside the group*, so it's important delete the right resource group name!

Azure CLI

```
# Get a list of resource groups in the active subscription
az group list --output table

# Delete a resource group and do not wait for the operation to finish
az group delete --name <msdocs-tutorial-rg-0000000> --no-wait
```

Tip

The `--yes` parameter of the `az group delete` command will by-pass the console confirmation prompt.

Delete multiple Azure resources using a script

When you're working with a large number of resources and you don't want to delete all the objects within a group, consider using a script. This example gets a list of all the Azure storage accounts created in this tutorial and deletes them in a for-each loop.

Azure CLI

```
# Set your resource group variable
resourceGroup=<msdocs-tutorial-rg-0000000>

# Get the name of all storage accounts in a resource group.
az storage account list --resource-group $resource-group \
    --query "[].{Name:name}" \
    --output table

# Delete storage accounts without a confirmation prompt.
for saList in $(az storage account list --resource-group $resource-group \
\ \
    --query "[?starts_with(name, 'msdocs') == `true`].id" \
    --output tsv); do
    echo "deleting storage account $saList"
    az storage account delete --ids $saList --yes
done

# Verify the storage accounts are gone.
az storage account list --resource-group $resource-group \
    --query "[?starts_with(name, 'msdocs') == `true`].name"
```

Get more details

Do you want more detail on one of the references used in this tutorial step? Use these links to learn more.

- [az group list](#)
- [az group delete](#)
- [az storage account list](#)
- [az storage account delete](#)

This is the end of the tutorial, and look at all you accomplished! You're now officially onboarded with the Azure CLI. Well done!

Create an Azure service principal with Azure CLI

Article • 06/05/2024

Automated tools that use Azure services should always have restricted permissions to ensure that Azure resources are secure. Therefore, instead of having applications sign in as a fully privileged user, Azure offers service principals. An Azure service principal is an identity created for use with applications, hosted services, and automated tools. This identity is used to access resources.

In this tutorial, you learn how to:

- ✓ Create a service principal
- ✓ Sign in using a service principal and password
- ✓ Sign in using a service principal and certificate
- ✓ Manage service principal roles
- ✓ Create an Azure resource using a service principal
- ✓ Reset service principal credentials

Prerequisites

- In a subscription, you must have `User Access Administrator` OR `Role Based Access Control Administrator` permissions, or higher, to create a service principal. For a list of roles available for Azure role-based access control (Azure RBAC), see [Azure built-in roles](#).
- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).

[Launch Cloud Shell](#)

- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

Create a service principal

Use the `az ad sp create-for-rbac` Azure CLI reference command to create a service principal. This example doesn't specify a `--name` parameter, so a name containing a time stamp is automatically created.

Azure CLI

```
az ad sp create-for-rbac
```

Output console:

Output

```
{
  "appId": "myAppId",
  "displayName": "myServicePrincipalName",
  "password": "myServicePrincipalPassword",
  "tenant": "myTenantId"
}
```

If you aren't adhering to resource naming conventions and plan to create a role and scope for your new service principal later, the `az ad sp create-for-rbac` command without parameters is an acceptable solution. However, without a role and scope, the new service principal doesn't have access to resources. It just exists.

When you create a service principal without parameters, also complete these steps:

- Record your system-assigned password as you can't retrieve it again. If you lose the password, reset it using [az ad sp credential reset](#) as explained in [Reset service principal credentials](#).

- Set the role assignment for your new service principal by using `az role assignment create` as explained in [Manage service principal roles](#).

Note

If your account doesn't have permission to create a service principal, `az ad sp create-for-rbac` returns an error message containing "Insufficient privileges to complete the operation". Contact your Microsoft Entra admin to create a service principal.

In a Microsoft Entra ID directory where user setting **Users can register applications** has been set to **No**, you must be a member of one of the following Microsoft Entra ID built-in roles (which have the action: `microsoft.directory/applications/createAsOwner` or `microsoft.directory/applications/create`):

- [Application Developer](#)
- [Application Administrator](#)
- [Cloud Application Administrator](#)
- [Global Administrator](#)
- [Hybrid Identity Administrator](#)

For more information about user settings in Microsoft Entra ID, see [Restrict who can create applications](#).

Create a service principal with role and scope

As a best practice, always assign a specific `--role` and `--scopes` when you create a service principal. Follow these steps:

- Determine the correct role.

When determining role, always use the principle of least privilege. For example, don't give your service principal `contributor` permissions to a subscription if the service principal only needs to access Azure storage within a resource group. Consider a specialize role like `storage blob data contributor`. For a complete list of available roles in Azure RBAC, see [Azure built-in roles](#).

- Get a value for the scopes parameter.

Find and copy the **Resource ID** of the Azure resource the new service principal needs to access. This information is usually found in the Azure portal's **Properties** or **Endpoints** page of each resource. Here are common `--scopes` examples, but *rely on your Resource ID for an actual format and value*.

 Expand table

Scope	Example
Subscription	<code>/subscriptions/mySubscriptionID</code>
Resource group	<code>/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName</code>
Virtual machine	<code>/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.Compute/virtualMachines/myVMname</code>
Storage account file service	<code>/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.Storage/storageAccounts/myStorageAccountName/file</code>
Data factory	<code>/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.DataFactory/factories/myDataFactoryName</code>

For more scope examples, see [Understand scope for Azure RBAC](#).

- Create the service principal.

In this example, a new service principal named `myServicePrincipalName1` is created with `reader` permissions to all resources in resource group `RG1`.

Bash

Azure CLI

```
# Bash script
az ad sp create-for-rbac --name myServicePrincipalName1 \
    --role reader \
    --scopes /subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myRG1
```

The `--scopes` parameter accepts a space-delimited list of scopes. In this example, a new service principal named `myServicePrincipalName2` is created with `reader` permissions to all resources in resource group `myRG1`. This service principal is also given `reader` permissions to `myVM` located in `myRG2`.

Azure CLI

```
# Bash script
az ad sp create-for-rbac --name myServicePrincipalName2 \
    --role reader \
    --scopes /subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myRG1
/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myRG2/providers/Microsoft.Compute/virtualMachines/myVM
```

If you decide that you granted too few or too many permissions to your new service principal, alter the permissions by [managing service principal roles](#).

Create a service principal using variables

You can also create a service principal using variables:

Bash

Azure CLI

```
# Bash script
let "randomIdentifier=$RANDOM*$RANDOM"
servicePrincipalName="msdocs-sp-$randomIdentifier"
roleName="azureRoleName"
subscriptionID=$(az account show --query id --output tsv)
# Verify the ID of the active subscription
echo "Using subscription ID $subscriptionID"
resourceGroup="myResourceGroupName"

echo "Creating SP for RBAC with name $servicePrincipalName, with role $roleName and in scopes
$subscriptionID/resourceGroups/$resourceGroup"
az ad sp create-for-rbac --name $servicePrincipalName \
    --role $roleName \
    --scopes $subscriptionID/resourceGroups/$resourceGroup
```

For a complete list of service principal properties, use `az ad sp list` and see [Get an existing service principal](#).

⚠ Warning

When you create an Azure service principal using the `az ad sp create-for-rbac` command, the output includes credentials that you must protect. Be sure that you do not include these credentials in your code or check the credentials into your source control. As an alternative, consider using [managed identities](#) if available to avoid the need to use credentials.

Next Steps

Now that you've learned how to create an Azure service principal, proceed to the next step to learn how to use service principals with password-based authentication.

[Use password-based authentication](#)

Use an Azure service principal with password-based authentication

Article • 06/05/2024

When creating a service principal, you choose the type of sign-in authentication it uses. There are two types of authentication available for Azure service principals: **password-based authentication** and **certificate-based authentication**. Password-based authentication is good to use when learning about service principals, but we recommend using [certificate-based authentication](#) for applications.

This step in the tutorial explains how to use a service principal password to access an Azure resource.

Create a service principal containing a password

The default behavior of [az ad sp create-for-rbac](#) is to create a service principal with a random password.

Azure CLI

```
az ad sp create-for-rbac --name myServicePrincipalName \
    --role reader \
    --scopes
    /subscriptions/mySubscriptionId/resourceGroups/myResourceGroupName
```

Output Console:

Output

```
{
  "appId": "myServicePrincipalId",
  "displayName": "myServicePrincipalName",
  "password": "myServicePrincipalPassword",
  "tenant": "myOrganizationTenantId"
}
```

The output for a service principal with password authentication includes the `password` key. **Make sure you copy this value** - it can't be retrieved. If you lose the password, [reset the service principal credentials](#).

Sign in using a service principal using a password

Test the new service principal's credentials and permissions by signing in. To sign in with a service principal, you need the `appId` (also known as "service principal ID", "username" or "assignee"), `tenant`, and `password`. Here's an example:

Azure CLI

```
az login --service-principal \
--username myServicePrincipalId \
--password myServicePrincipalPassword \
--tenant myOrganizationTenantID
```

If you don't know your `appId` or `--tenant`, retrieve it by using the `az ad sp list` command.

Azure CLI

```
spID=$(az ad sp list --display-name myServicePrincipalName --query "[].{spID:appId}" --output tsv)
tenantID=$(az ad sp list --display-name myServicePrincipalName --query "[].{tenantId:appOwnerOrganizationId}" --output tsv)
echo "Using appId $spID in tenant $tenantID"

az login --service-principal \
--username $spID \
--password {paste your password here} \
--tenant $tenantID
```

If you're testing in an organization that requires two-factor authentication, error message "...Interactive authentication is needed..." is displayed. As an alternative, use a certificate or [managed identities](#).

ⓘ Important

If you want to avoid displaying your password on console and are using `az login` interactively, use the `read -s` command in bash.

Bash

```
read -sp "Azure password: " AZ_PASS && echo && az login --service-principal -u <app-id> -p $AZ_PASS --tenant <tenant>
```

In PowerShell, use the `Get-Credential` cmdlet.

PowerShell

```
$AzCred = Get-Credential -UserName <app-id>
az login --service-principal -u $AzCred.UserName -p
$AzCred.GetNetworkCredential().Password --tenant <tenant>
```

Next Steps

Now that you've learned how to work with service principals using a password, proceed to the next step to learn how to use service principals with certificate-based authentication.

[Use certificate-based authentication](#)

Use an Azure service principal with certificate-based authentication

Article • 06/05/2024

When creating a service principal, you choose the type of sign-in authentication it uses. There are two types of authentication available for Azure service principals: **password-based authentication** and **certificate-based authentication**.

We recommend using certificate-based authentication due to the security restrictions of password-based authentication. Certificate-based authentication enables you to adopt a phishing resistant authentication by using [conditional access policies](#), which better protects Azure resources. To learn more about why certificate-based authentication is more secure, see [Microsoft Entra certificate-based authentication](#).

This step in the tutorial explains how to use a service principal certificate to access an Azure resource.

Create a service principal containing a new certificate

To create a *self-signed* certificate for authentication, use the `--create-cert` parameter:

```
Azure CLI  
  
az ad sp create-for-rbac --name myServicePrincipalName \  
    --role roleName \  
    --scopes  
    /subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName \  
    --create-cert
```

Console output:

```
Output  
  
{  
  "appId": "myServicePrincipalID",  
  "displayName": "myServicePrincipalName",  
  "fileWithCertAndPrivateKey": "certFilePath\certFileName.pem",  
  "password": null,  
  "tenant": "myOrganizationTenantID"  
}
```

Unless you store the certificate in Key Vault, the output includes the `fileWithCertAndPrivateKey` key. This key's value tells you where the generated certificate is stored. Copy the certificate to a secure location. If you lose access to a certificate's private key, [reset the service principal credentials](#).

The contents of a PEM file can be viewed with a text editor. Here's a PEM file example:

-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggsjAgEAAoIBAQDLhs+mr3SAkR9r
dGh8nPh3AJ/jynWNMQRaeCPhUH99eQiYI1DK7f6AjwFB9z+57W6HoPeFHFNt0/v
1BmCOCdwXptlV3W48V2hAK2Rwmhtxyo3RjJhsqw6mltpdtHv3zYbLpquQsyspRi3
uVGrt8UW2jV3e8py4VzOhqHy+enE4xHLSdtpmG401XWNNUv3YcET/qSutYieM12JV
'LXdZS...0`" `xUS3` dt i9
-----END PRIVATE KEY-----
-----BEGIN CERTIFICATE-----
MIICoTCCAYkCAgPoMA0GCSqGSIb3DQEBBQUAMBQxEjAQBgnNVBAMMCUNMSS1Mb2dp
bjAiGA8yMDIzMDkyOTIxMjkyMVoYDzIwMjQwOTI5MjEyOTIzWjAUMRIwEAYDVQQD
DA1DTEktTG9naW4wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDLhs+m
r3SAkR9rdGh8nPh3AJ/jynWNMQRaeCPhUH99eQiYI1DK7f6AjwFB9z+57W6HoPeF
-----END CERTIFICATE-----
-----END CERTIFICATE-----

Create a service principal using an existing certificate

Create a service principal with an existing certificate by using the `--cert` parameter. Any tool that uses this service principal must have access to the certificate's private key. Certificates should be in an ASCII format such as PEM, CER, or DER. Pass the `certificate` as a string, or use the `@path` format to load the certificate from a file.

When you use a PEM file, the **CERTIFICATE** must be appended to the **PRIVATE KEY** within the file.

```
...
-----END CERTIFICATE-----"
```

Azure CLI

```
# create a service principal with the certificate file location
az ad sp create-for-rbac --name myServicePrincipalName \
    --role roleName \
    --scopes
/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName \
    --cert @/path/to/cert.pem
```

Work with Azure Key Vault

The `--keyvault` parameter can be added to create or retrieve certificates in Azure Key Vault. When you use the `--keyvault` parameter, the `--cert` parameter is also required. In this example, the `--cert` value is the name of the certificate.

Azure CLI

```
# Create a service principal storing the certificate in Azure Key Vault
az ad sp create-for-rbac --name myServicePrincipalName \
    --role roleName \
    --scopes
/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName \
    --create-cert \
    --cert myCertificateName \
    --keyvault myVaultName
```

Azure CLI

```
# Create a service principal using an existing certificate in Azure Key Vault
az ad sp create-for-rbac --name myServicePrincipalName \
    --role roleName \
    --scopes
/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName \
    --cert myCertificateName \
    --keyvault myVaultName
```

Retrieve a certificate from Azure Key Vault

For a certificate stored in Azure Key Vault, retrieve the certificate with its private key with [az keyvault secret show](#) and convert it to a PEM file. In Azure Key Vault, the name of the certificate's secret is the same as the certificate name.

Azure CLI

```
az keyvault secret download --file /path/to/cert.pfx \
    --vault-name VaultName \
    --name CertName \
    --encoding base64
openssl pkcs12 -in cert.pfx -passin pass: -passout pass: -out cert.pem -nodes

az login --service-principal -u "<myAppClientID>" -p cert.pem --tenant "<myTenantID>"
```

Convert an existing PKCS12 file

If you already have a PKCS#12 file, you can convert it to PEM format using OpenSSL. If you have a password, change the `passin` argument.

Console

```
openssl pkcs12 -in fileName.p12 -clcerts -nodes -out fileName.pem -passin
pass: -passout pass:
```

Append a certificate to a service principal

Use the `--append` parameter in `az ad sp credential reset` to append a certificate to an existing service principal. By default, this command clears all passwords and keys so use carefully.

Azure CLI

```
az ad sp credential reset --id myServicePrincipalID \
    --append \
    --cert @/path/to/cert.pem
```

Console output:

Output

```
Certificate expires yyyy-mm-dd hh:mm:ss+00:00. Adjusting key credential end
date to match.

The output includes credentials that you must protect. Be sure that you do
not include these credentials in your code or check the credentials into
your source control. For more information, see https://aka.ms/azadsp-cli
{
    "appId": "myServicePrincipalID",
```

```
"password": null,  
"tenant": "myOrganizationTenantID"  
}
```

Sign in with a service principal using a certificate

To sign in with a certificate, the certificate must be available locally as a PEM or DER file in ASCII format. PKCS#12 files (.p12/.pfx) don't work. When you use a PEM file, the **PRIVATE KEY** and **CERTIFICATE** must be appended together within the file. You don't need to prefix the path with an @ like you do with other az commands.

Azure CLI

```
az login --service-principal \  
    --username myServicePrincipalID \  
    --tenant myOwnerOrganizationId \  
    --password /path/to/cert
```

Next Steps

Now that you've learned how to work with service principals using a certificate, proceed to the next step to learn how to retrieve an existing service principal.

[Get an existing service principal](#)

Get an existing service principal

Article • 06/05/2024

List service principals

If you already have an existing service principal that you wish to use, this step explains how to retrieve your existing service principal.

A list of the service principals in a tenant can be retrieved with `az ad sp list`. By default this command returns the first 100 service principals for your tenant. To get all of a tenant's service principals, use the `--all` parameter. Getting this list can take a long time, so it's recommended that you filter the list with one of the following parameters:

- `--display-name` requests service principals that have a *prefix* that match the provided name. The display name of a service principal is the value set with the `--name` parameter during creation. If you didn't set `--name` during service principal creation, the name prefix is `azure-cli-`.
- `--spn` filters on exact service principal name matching. The service principal name always starts with `https://`. If the value you used for `--name` wasn't a URI, this value is `https://` followed by the display name.
- `--show-mine` requests only service principals created by the signed-in user.
- `--filter` takes an OData filter, and performs *server-side* filtering. This method is recommended over filtering client-side with the CLI's `--query` parameter. To learn about OData filters, see [OData expression syntax for filters](#).

The information returned for service principal objects is verbose. To get only the information necessary for sign-in, use the query string `[].{id:appId, tenant:appOwnerOrganizationId}`. Here's an example that gets the sign-in information for all service principals created by the currently logged in user:

Azure CLI

```
az ad sp list --show-mine --query "[].{SPname:displayName, SPid:appId, tenant:appOwnerOrganizationId}" --output table
```

If you're working in a large organization with many service principals, try these command examples:

Azure CLI

```
# get service principals containing a keyword
az ad sp list --display-name mySearchWord --output table

# get service principals using an OData filter
az ad sp list --filter "displayname eq 'myExactServicePrincipalName'" --
output json

# get a service principal having a certain servicePrincipalNames property
value
az ad sp list --spn https://spURL.com
```

ⓘ Important

The user and tenant can both be retrieved with [az ad sp list](#) and [az ad sp show](#), but authentication secrets or the authentication method is not available. Secrets for certificates in Azure Key Vault can be retrieved with [az keyvault secret show](#), but no other secrets are stored by default. If you forget an authentication method or secret, [reset the service principal credentials](#).

Service principal properties

When you get a list of service principals using `az ad sp list`, there are many output properties you can reference in your script.

Output

```
[  
 {  
   "accountEnabled": true,  
   "addIns": [],  
   "alternativeNames": [],  
   "appDescription": null,  
   "appDisplayName": "myServicePrincipalName",  
   "appId": "00000000-0000-0000-0000-000000000000",  
   "appOwnerOrganizationId": "00000000-0000-0000-0000-000000000000",  
   "appRoleAssignmentRequired": false,  
   "appRoles": [],  
   "applicationTemplateId": null,  
   "createdDateTime": null,  
   "deletedDateTime": null,  
   "description": null,  
   "disabledByMicrosoftStatus": null,  
   "displayName": "myServicePrincipalName",  
   "homepage": "https://myURL.com",  
   "id": "00000000-0000-0000-000000000000",  
   "info": {  
     "logoUrl": null,
```

```
        "marketingUrl": null,
        "privacyStatementUrl": null,
        "supportUrl": null,
        "termsOfServiceUrl": null
    },
    "keyCredentials": [],
    "loginUrl": null,
    "logoutUrl": null,
    "notes": null,
    "notificationEmailAddresses": [],
    "oauth2PermissionScopes": [
        {
            "adminConsentDescription": "my admin description",
            "adminConsentDisplayName": "my admin display name",
            "id": "00000000-0000-0000-0000-000000000000",
            "isEnabled": true,
            "type": "User",
            "userConsentDescription": "my user description",
            "userConsentDisplayName": "my user display name",
            "value": "user_impersonation"
        }
    ],
    "passwordCredentials": [],
    "preferredSingleSignOnMode": null,
    "preferredTokenSigningKeyThumbprint": null,
    "replyUrls": [],
    "resourceSpecificApplicationPermissions": [],
    "samlSingleSignOnSettings": null,
    "servicePrincipalNames": [
        "00000000-0000-0000-0000-000000000000",
        "https://myURL.com"
    ],
    "servicePrincipalType": "Application",
    "signInAudience": null,
    "tags": [
        "WindowsAzureActiveDirectoryIntegratedApp"
    ],
    "tokenEncryptionKeyId": null,
    "verifiedPublisher": {
        "addedDateTime": null,
        "displayName": null,
        "verifiedPublisherId": null
    }
}
]
```

Use the `--query` parameter to retrieve and store service principal properties in variables.

Bash

Azure CLI

```
# Bash script
spID=$(az ad sp list --display-name myServicePrincipalName --query "[].{spID:appId}" --output tsv)
tenantID=$(az ad sp list --display-name myServicePrincipalName --query "[].{tenant:appOwnerOrganizationId}" --output tsv)
userConsentDescr=$(az ad sp list --display-name myServicePrincipalName --query "[].{ucs:oauth2PermissionScopes.userConsentDescription[0]}" --output tsv)
echo "Using appId $spID in tenant $tenantID for $userConsentDescr"
```

Next Steps

Now that you've learned how to retrieve your existing service principal, proceed to the next step to learn how to manage your service principal roles.

[Manage service principal roles](#)

Manage service principal roles

Article • 06/05/2024

In order to restrict access to your Azure resources, you can use a service principal to manage role assignments. Each role provides different permissions allowed by the user when accessing Azure resources. This step in the tutorial explains how to create and remove service principal roles.

The Azure CLI has the following commands to manage role assignments:

- [az role assignment list](#)
- [az role assignment create](#)
- [az role assignment delete](#)

Create or remove a role assignment

The **Contributor** role has full permissions to read and write to an Azure account. The **Reader** role is more restrictive with read-only access. Always use the principle of least privilege. For a complete list of available roles in Azure RBAC, see [Azure built-in roles](#).

Adding a role *doesn't* restrict previously assigned permissions. This example adds the **Reader** role and removes the **Contributor** role:

Azure CLI

```
az role assignment create --assignee myServicePrincipalID \
    --role Reader \
    --scope /subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName

az role assignment delete --assignee myServicePrincipalID \
    --role Contributor \
    --scope /subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName
```

Output Console:

Output

```
{
  "condition": null,
  "conditionVersion": null,
  "createdBy": null,
  "createdOn": "yyyy-mm-ddT00:00:00.000000+00:00",
  "delegatedManagedIdentityResourceId": null,
  "description": null,
  "id": "/subscriptions/00000000-0000-0000-0000-000000000000/providers/Microsoft.Authorization/roleAssignments/00000000-
0000-0000-000000000000",
  "name": "00000000-0000-0000-0000-000000000000",
  "principalId": "00000000-0000-0000-0000-000000000000",
  "principalType": "ServicePrincipal",
  "roleDefinitionId": "/subscriptions/00000000-0000-0000-0000-
000000000000/providers/Microsoft.Authorization/roleDefinitions/00000000-0000-0000-0000-000000000000",
  "scope": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myResourceGroupName",
  "type": "Microsoft.Authorization/roleAssignments",
  "updatedBy": "00000000-0000-0000-0000-000000000000",
  "updatedOn": "yyyy-mm-ddT00:00:00.000000+00:00"
}
```

How to get a value for the scope parameter

One question you might have is "How do I know the `--scope` parameter value?" The answer is to find and copy the Resource ID of the Azure resource your service principal needs to access. This information is usually found in the Azure portal's **Properties** or **Endpoints** page of each resource. Here are common `--scope` examples, but *rely on your Resource ID for an actual format and value*.

 Expand table

Scope	Example
Subscription	/subscriptions/mySubscriptionID
Resource	/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName

Scope	Example
group	
Virtual machine	/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.Compute/virtualMachines/myVMname
Storage account file service	/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.Storage/storageAccounts/myStorageAccountName/fileService
Data factory	/subscriptions/mySubscriptionID/resourceGroups/myResourceGroupName/providers/Microsoft.DataFactory/factories/myDataFactoryName

For more scope examples, see [Understand scope for Azure RBAC](#).

Verify changes

The changes can be verified by listing the assigned roles:

Azure CLI

```
# list all role assignments for the current subscription
az role assignment list --output table

# list role assignments for a user
az role assignment list --assignee myUserName@contoso.com

# list role assignments for a subscription
az role assignment list --subscription mySubscriptionID
```

You can also go into the [Azure portal](#) and manually assign the role to the service principal from the Access control (IAM) menu. For more examples on listing role assignments, see [List Azure role assignments using Azure CLI](#).

Next Steps

Now that you've learned how to manage your service principal roles, proceed to the next step to learn how to use service principals to create a resource.

[Create a resource using service principal](#)

Create a resource using a service principal

Article • 06/05/2024

If given the necessary permissions, a service principal can create and manage Azure resources just like an account. This tutorial step provides an example of how to create a resource for [Azure Storage](#) using a service principal and the following commands:

- [az login](#)
- [az group create](#)
- [az storage account create](#)
- [az storage account keys list](#)

To sign in with a service principal, you need the `appID`, `tenantID`, and `password` returned in the console output when you [created a service principal](#).

1. Sign in as the service principal.

Azure CLI

```
az login --service-principal \
    --username myServicePrincipalID \
    --password myServicePrincipalPassword \
    --tenant myOrganizationTenantID
```

Output console:

Output

```
[  
  {  
    "cloudName": "AzureCloud",  
    "homeTenantId": "tenantID",  
    "id": "mySubscriptionId",  
    "isDefault": true,  
    "managedByTenants": [],  
    "name": "mySubscriptionName",  
    "state": "Enabled",  
    "tenantId": "tenantID",  
    "user": {  
      "name": "myServicePrincipalID",  
      "type": "servicePrincipal"  
    }  
  }  
]
```

2. Create a resource group to hold all resources for the same project.

```
Azure CLI
```

```
az group create --location westus --name myResourceGroupName
```

3. Create a storage account.

For Azure Storage, valid values for the <KIND> parameter are:

- BlobStorage
- BlockBlobStorage
- FileStorage
- Storage
- StorageV2

```
Azure CLI
```

```
az storage account create --name myStorageAccountName \
--resource-group myResourceGroupName \
--kind <KIND> \
--sku F0 \
--location westus \
--yes
```

4. Get resource keys, which you use in your code to authenticate to the Azure storage account.

```
Azure CLI
```

```
az storage account keys list --resource-group myResourceGroupName \
--account-name myStorageAccountName
```

Output Console:

```
Output
```

```
[
  {
    "creationTime": "2023-09-15T17:29:49.554030+00:00",
    "keyName": "key1",
    "permissions": "FULL",
    "value": "myKeyValue1"
  },
  {
    "creationTime": "2023-09-15T17:29:49.554030+00:00",
    "keyName": "key2",
```

```
    "permissions": "FULL",
    "value": "myKeyValue2"
}
]
```

Next Steps

Now that you've learned how to create a resource using service principal, proceed to the next step to learn how to reset your service principal credentials.

[Reset service principal credentials](#)

Reset service principal credentials

Article • 07/30/2024

If you lose your service principal credentials, reset the credentials using [az ad sp credential reset](#). This step details how to reset your service principal password or certificate.

⚠️ Warning

While learning to manage Azure service principals, returning passwords and credential locations in your terminal, and consequently in your log file, is common. However, when outside of a testing environment, [store credential output in a variable](#).

Reset credentials returning output to the console and log file

Use [az ad sp credential reset](#) command to create a new password or certificate for your service principal. If you don't know the ID associated with your service principal, use the [az ad sp list](#) command as explained in [Get an existing service principal](#).

Reset a service principal password.

Azure CLI

```
az ad sp credential reset --id myServicePrincipalID
```

Console output showing the new password in the console. This information is also written in the log.

Output

```
{
  "appId": "myServicePrincipalID",
  "password": "myServicePrincipalNewPassword",
  "tenant": "myTenantID"
}
```

Reset a service principal credential with a new self-signed certificate.

Azure CLI

```
az ad sp credential reset --id myServicePrincipalID --create-cert
```

Console output showing the new certificate location in the console. This information is also written in the log.

Output

```
{  
    "appId": "myServicePrincipalID",  
    "fileWithCertAndPrivateKey": "myLocation/myPemFileName.pem",  
    "password": null,  
    "tenant": "myTenantID"  
}
```

Reset credentials storing output in a variable

To avoid storing credentials in your log file, use the `--query` parameter to store output in a variable. When testing, use the `echo` command to see the value of your variable, but understand that `echo` writes to the log.

Reset a service principal credential with a password.

Bash

Azure CLI

```
# Bash script  
myNewPassword=$(az ad sp credential reset --id myServicePrincipalID --  
query password --output tsv)  
  
# the echo command writes to the log file  
# only use it when testing  
echo $myNewPassword
```

Next Steps

Now that you've learned how to reset your service principal credentials, proceed to the next step to see how to clean up tutorial resources.

[Cleanup & troubleshoot service principals](#)

 Collaborate with us on
GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)

Cleanup & troubleshoot service principals

Article • 06/05/2024

Congratulations! You learned how to create, retrieve, and work with service principals! Now that you have completed the tutorial, it's time to clean up the created service principal resources.

Cleanup service principal resources

The safest way to remove all resources used in this tutorial is to use [az group delete](#). The `--no-wait` parameter keeps the CLI from blocking while the deletion takes place.

Azure CLI

```
az group delete --name myResourceGroup --no-wait
```

If you prefer, delete individual service principals with the [az ad sp delete](#) command.

Azure CLI

```
az ad sp delete --id myServicePrincipalID
```

Troubleshoot service principals

Although you have completed the tutorial, you may still have questions regarding service principals that can be answered in this section.

Insufficient privileges

If your account doesn't have permission to create a service principal, `az ad sp create-for-rbac` returns an error message containing "Insufficient privileges to complete the operation." Contact your Microsoft Entra administrator to obtain `User Access Administrator` or `Role Based Access Control Administrator` permissions.

Invalid tenant

If you have specified an invalid subscription ID, you see the error message "The request didn't have a subscription or a valid tenant level resource provider." If using variables, use the Bash `echo` command to see the value being passed to the reference command. Use `az account set` to change your subscription or learn [How to manage Azure subscriptions with the Azure CLI](#).

Resource group not found

If you have specified an invalid resource group name, you see the error message "Resource group 'name' couldn't be found." If using variables, use the Bash `echo` command to see the value being passed to both the subscription and reference commands. Use `az group list` to see the resource groups for the current subscription, or learn [How to manage Azure resource groups with the Azure CLI](#).

Authorization to perform action

If your account doesn't have permission to assign a role, you see an error message that your account "does not have authorization to perform action 'Microsoft.Authorization/roleAssignments/write'." Contact your Microsoft Entra administrator to manage roles.

Interactive authentication is needed

When signing in with password authentication, error message "...Interactive authentication is needed..." occurs if your organization requires multifactor authentication. Switch to certificate-based authentication, or consider using [managed identities](#).

Compliant device required

If you try to create a service principal using a device that is not compliant with your organizations access policies, you will receive message "...Conditional Access policy requires a compliant device...". Switch to a computer that meets your organization's access policies.

See also

- Application and service principal objects in Microsoft Entra ID
- Create a service principal using the Azure portal
- Create an Azure service principal with Azure PowerShell

Overview and Prerequisites

Article • 06/05/2024

In this tutorial, you learn how to create a virtual network (VNet) and deploy a virtual machine (VM) to the VNet with the Azure CLI. This tutorial also covers Azure CLI specific concepts such as shell variables and output queries.

This tutorial can be completed with the interactive experience offered through Azure Cloud Shell, or you may [install the CLI](#) locally.

Use **ctrl-shift-v** (**cmd-shift-v** on macOS) to paste tutorial text into Azure Cloud Shell.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
A blue rectangular button with a white 'A' icon on the left, the text 'Launch Cloud Shell' in the center, and a small copy icon on the right.
A blue rectangular button with a white 'A' icon on the left, the text 'Launch Cloud Shell' in the center, and a small copy icon on the right.
- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).

Shell variables

Shell variables store values for future use and can be used to pass values to command parameters. Shell variables allow for the reuse of commands, both on their own and in scripts. This tutorial uses shell variables for easier customization of command parameters. To use your own parameter values instead of using the provided values, change the values assigned to the shell variables. For more information about shell variables, see [Use shell variables](#).

Create a resource group

In Azure, all resources are allocated in a resource management group. Resource groups provide logical groupings of resources that make them easier to work with as a collection. Use the `az group create` command to create a resource group named `VMTutorialResources`.

Bash

```
Azure CLI

# create Bash shell variables
resourceGroup=VMTutorialResources
location=eastus

az group create --name $resourceGroup --location $location
```

Create a resource group in interactive mode

Interactive mode offers new AI functionalities that allow the user to run and search for commands more efficiently. To get a better understanding of required parameters and steps related to `az group create`, try out Azure CLI in interactive mode by running the `az interactive` command. Please note that all commands throughout this tutorial can be run in interactive mode.

Azure CLI

```
# install interactive mode
az interactive
```

In the following example, the command recommendation feature in interactive mode lists and defines all of the required parameters needed for `az group create`:

Azure CLI

```
# get options for az group create
az>> az group create
```

```
Windows PowerShell * + - □ ×
A new Recommender is added which can make the completion ability more intelligent and provide the scenario completion!
If you want to disable this feature, you can use 'az config set interactive.enable_recommender=False' to disable it.

A command preload mechanism was added to prevent lagging and command run errors.
You can skip preloading in a single pass by CTRL+C or turn it off by setting 'az config set interactive.enable_preloading=False'

Loading command table... Expected time around 1 minute.

az>> az group create
--location      [Required] Location. Values from: `az account list-locations`. You can configure the default location using ^J..
--name          [Required] name of the new resource group
--resource-group [Required] name of the new resource group
--managed-by    The ID of the resource that manages this resource group.
--tags          space-separated tags: key[=value] [key[=value] ...]. Use "" to clear existing tags.

Create a new resource group. *
[1] Create a new resource group in the West US region.
az group create -l westus -n MyResourceGroup
CTRL+Y (^) CTRL+N (v)

/[keyword]      : search for commands and scenarios
#[cmd]         : use commands outside the application
::[num]        : complete a recommended scenario step by step
[cmd][param]??[query]: Inject jmespath query from previous command
??[query]       : Jmespath query of the previous command
[cmd]:::[num]   : do a step by step tutorial of example
$              : get the exit code of the previous command
%[cmd]         : set a scope, and scopes can be chained with spaces
%..            : go back a scope
[F1]Layout [F2]Defaults [F3]Keys [Space]Predict [Ctrl+C]Clear Screen [Ctrl+D]Quit Subscription:
```

Create a virtual network

Article • 06/05/2024

Virtual networks (VNets) allow virtual machines (VMs) and other Azure resources to communicate securely with each other, the internet, and on-premises networks. VNets can also be connected to other VNets if their address ranges don't overlap. In this section, you learn how to create a virtual network with a subnet.

Subnets allow you to segment the VNet address space into subnetworks for organization purposes. Azure deploys resources to subnets within a virtual network, so you need to create a subnet.

Use the [az network vnet create](#) command to create a virtual network named `TutorialVNet1` with address prefix of 10.0.0.0/16 and a subnet named `TutorialSubnet1` with address prefix of 10.0.0.0/24.

Bash

Azure CLI

```
# create Bash shell variables
vnetName=TutorialVNet1
subnetName=TutorialSubnet1
vnetAddressPrefix=10.0.0.0/16
subnetAddressPrefix=10.0.0.0/24

# Use the existing resource group
resourceGroup=VMTutorialResources

az network vnet create \
--name $vnetName \
--resource-group $resourceGroup \
--address-prefixes $vnetAddressPrefix \
--subnet-name $subnetName \
--subnet-prefixes $subnetAddressPrefix
```

Create a virtual machine on a virtual network

Article • 06/05/2024

Virtual machines (VM) in Azure have a large number of dependencies. The CLI creates these resources for you based on the command-line arguments you specify. In this section, you learn how to deploy a VM to a VNet.

To deploy a VM on a VNet, they must have the same Azure location. Once a VM is created, you can't change the VNet to which it's connected.

Create a VM

Use the [az vm create](#) command to create a new virtual machine running Ubuntu. This virtual machine uses SSH authentication for sign in, and is connected to the subnet and VNet you created in the previous section. In PowerShell, declare variables with a "\$", and use quotes for values, such as \$vmName = "TutorialVM1".

Bash

```
Azure CLI

# create Bash shell variable
vmName=TutorialVM1

az vm create \
--resource-group $resourceGroup \
--name $vmName \
--image Ubuntu2204 \
--vnet-name $vnetName \
--subnet $subnetName \
--generate-ssh-keys \
--output json \
--verbose
```

ⓘ Note

If you have an SSH key named `id_rsa` already available, this key is used for authentication rather than having a new key generated.

As the VM is created, you see the local values used and Azure resources being created due to the `--verbose` option. Once the VM is ready, a JSON is returned from the Azure service including the public IP address.

```
Output

{
  "fqdns": "",
  "id": "...",
  "location": "eastus",
  "macAddress": "...",
  "powerState": "VM running",
  "privateIpAddress": "...",
  "publicIpAddress": "<PUBLIC_IP_ADDRESS>",
  "resourceGroup": "TutorialResources",
  "zones": ""
}
```

Confirm that the VM is running by connecting over SSH.

```
Bash

ssh <PUBLIC_IP_ADDRESS>
```

Go ahead and sign out from the VM by typing `exit`.

There are other ways to get this IP address after the VM has started. In the next section, you'll see how to get detailed information on the VM, and how to filter it.

Get VM information with queries

Article • 06/05/2024

Now that a VM has been created, detailed information about it can be retrieved. The common command for getting information from a resource is `show`.

Azure CLI

```
az vm show --name $vmName --resource-group $resourceGroup
```

A Virtual Machine has many properties that can be difficult to parse visually. The returned JSON contains information on authentication, network interface storage, and more. Most importantly, it contains the Azure object IDs for resources that the VM is connected to. Object IDs allow accessing these resources directly to get more information about the VM's configuration and capabilities.

In order to extract the object ID, the `--query` argument is used. Queries are written in the [JMESPath query language](#). Start with getting the network interface controller (NIC) object ID.

Bash

Azure CLI

```
az vm show --name $vmName \
--resource-group $resourceGroup \
--query 'networkProfile.networkInterfaces[].id' \
--output tsv
```

There's a lot going on here, just by adding the query. Each part of it references a key in the output JSON, or is a JMESPath operator.

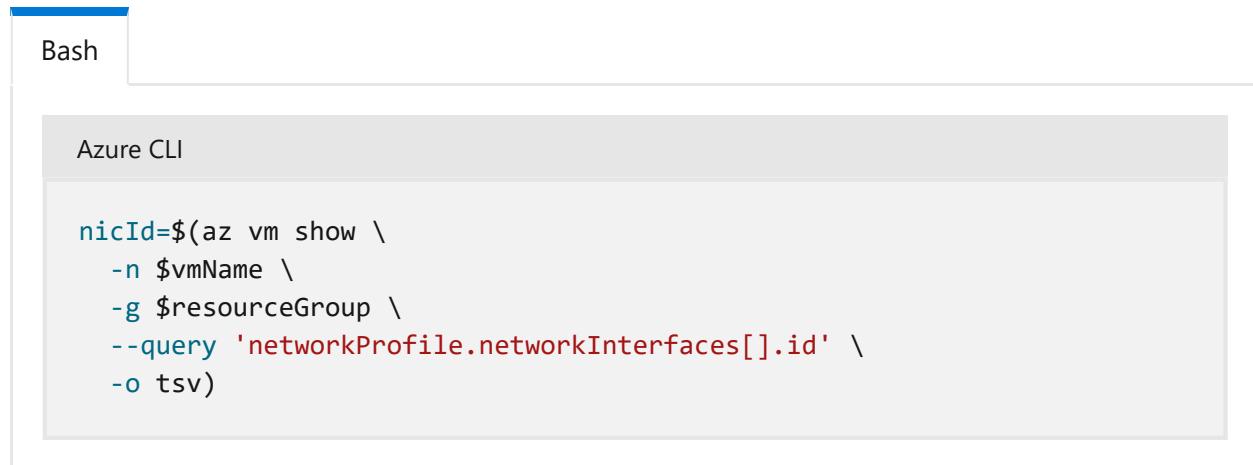
- `networkProfile` is a key of the top-level JSON, which has `networkInterfaces` as a subkey. If a JSON value is a dictionary, its keys are referenced from the parent key with the `.` operator.
- The `networkInterfaces` value is an array, so it's flattened with the `[]` operator. This operator runs the remainder of the query on each array element. In this case, it gets the `id` value of every array element.

The output format `tsv` (tab-separated values) is guaranteed to only include the result data and whitespace consisting of tabs and newlines. Since the returned value is a single

bare string, it's safe to assign directly to a shell variable.

For more information about querying Azure CLI output, see [How to query Azure CLI command output using a JMESPath query](#)

Go ahead and assign the NIC object ID to a shell variable now.



The screenshot shows a terminal window with a blue header bar. The header bar has a small blue icon on the left and the word "Bash" on the right. Below the header is a light gray toolbar with the text "Azure CLI". The main area of the terminal is a light gray text box containing the following command:

```
nicId=$(az vm show \
    -n $vmName \
    -g $resourceGroup \
    --query 'networkProfile.networkInterfaces[].id' \
    -o tsv)
```

This example also demonstrates the use of short arguments. You may use `-g` instead of `--resource-group`, `-n` instead of `--name`, and `-o` instead of `--output`.

Set shell variables from CLI output

Article • 06/05/2024

Now that you have the NIC ID, run `az network nic show` to get its information. You don't need a resource group here, since the resource group name is contained within the Azure resource ID.

Azure CLI

```
az network nic show --ids $nicId
```

This command shows all of the information for the network interface of the VM. This data includes DNS settings, IP information, security settings, and the MAC address. The following query shows how to obtain the public IP address and subnet object IDs.

Bash

Azure CLI

```
az network nic show --ids $nicId \
    --query '{IP:ipConfigurations[].publicIPAddress.id,
Subnet:ipConfigurations[].subnet.id}' \
    -o json
```

Output

```
{
  "IP": [
    "/subscriptions/.../resourceGroups/TutorialResources/providers/Microsoft.Network/publicIPAddresses/TutorialVM1PublicIP"
  ],
  "Subnet": [
    "/subscriptions/.../resourceGroups/TutorialResources/providers/Microsoft.Network/virtualNetworks/TutorialVM1VNET/subnets/TutorialVM1Subnet"
  ]
}
```

This command displays a JSON object that has custom keys ('IP' and 'Subnet') for the extracted values. While this style of output might not be useful for command-line tools, it helps with human readability and can be used with custom scripts.

In order to use command-line tools, change the command to remove the custom JSON keys and output as `tsv`. The `read` command processes this style of output by loading results into multiple variables. Since two values are displayed on separate lines, the `read` command delimiter must be set to the empty string rather than the default of non-newline whitespace.

Bash

Azure CLI

```
read -d '' ipId subnetId <<< $(az network nic show \
--ids $nicId \
--query '[ipConfigurations[].publicIPAddress.id,
ipConfigurations[].subnet.id]' \
-o tsv)
```

If you don't want to use the Bash `read` or PowerShell `-split` command, you can set each variable separately.

Azure CLI

```
$ipId = az network nic show --ids $nicId --query
'[ipConfigurations[].publicIPAddress.id]' -o tsv
$subnetId = az network nic show --ids $nicId --query
'[ipConfigurations[].subnet.id]' -o tsv
```

Use the public IP object ID to look up the public IP address and store it in a shell variable. The subnet ID was used to demonstrate how to query and store multiple values in the Azure CLI. Therefore, it isn't needed for the rest of the tutorial.

Bash

Azure CLI

```
vmIpAddress=$(az network public-ip show --ids $ipId \
--query ipAddress \
-o tsv)
```

Now you have the IP address of the VM stored in a shell variable. Go ahead and check that it's the same value that you used to initially connect to the VM.

Bash

```
echo $vmIpAddress
```

Cleanup

Article • 06/05/2024

Now that the tutorial is complete, it's time to clean up the created resources. You can delete individual resources with the `delete` command, but the safest way to remove all resources in a resource group is with `group delete`.

Azure CLI

```
az group delete --name $resourceGroup --no-wait
```

This command deletes the resources created during the tutorial, and is guaranteed to deallocate them in the correct order. The `--no-wait` parameter keeps the CLI from blocking while the deletion takes place. If you want to wait until the deletion is complete or watch it progress, use the `group wait` command.

Azure CLI

```
az group wait --name $resourceGroup --deleted
```

With cleanup completed, the tutorial is finished. Continue on for a summary of everything you learned and links to resources that will help you with your next steps.

Summary

Article • 06/05/2024

Congratulations! You learned how to create a VM that's connected to a VNet, used the `-query` and `--output` arguments to capture data to be stored in shell variables, and looked at some of the resources that get created for Azure VMs.

Where you go from here depends on what you plan to use the CLI for. There are various materials that go further in depth on the features covered in this tutorial.

Samples

If you want to get started right away with specific tasks, look at some sample scripts.

- [A - Z list of Azure CLI samples](#)
- [Azure CLI samples GitHub repository - VMs ↗](#)

Get support

If you'd like to give feedback, suggestions, or ask questions about the CLI, there are many ways for you to get in touch.

- `az feedback` is a built-in command for the CLI that allows providing free-form feedback to the team.
- For personalized support, see [Create an Azure support ticket in Azure CLI](#)
- File a feature request or a bug report with the CLI in the [Azure CLI repository ↗](#).
- Ask a question or get clarification by filing an issue in the [Azure CLI documentation repository ↗](#).

See also

For more information on the CLI features that were shown in the tutorial, see:

- [Output formats for Azure CLI commands](#)
- [How to query Azure CLI command output](#)
- [How to use variables in Azure CLI commands](#)

For more information on virtual networks and virtual machines, see:

- [Virtual networks and virtual machines in Azure](#)
- [Linux virtual machines in Azure](#)

- Bringing and creating Linux images in Azure

Azure CLI sample scripts

Article • 06/01/2024

Azure CLI samples provide end-to-end scenarios for jobs to be done. This article provides an A - Z list of Azure CLI samples written for Bash environments.

Not every Azure CLI reference command is used in a sample script. For a complete list of Azure CLI commands, see the [A - Z reference list](#). For a list of popular conceptual content for each reference group, see [Azure CLI conceptual article list](#).

List of samples

Use these three tabs to find Azure CLI samples. [List by subject area](#) and [List by reference group](#) tabs contain the same samples. Only the sort is different. The [Azure CLI samples GitHub repository](#) tab has more samples as not every GitHub sample is used in an article.

List by subject area

Use this list of Azure CLI samples to find samples for the subject area that *published* the script. (Many Azure CLI scripts use reference commands from more than one Azure service.)

[] Expand table

Subject area	Sample name
application-gateway	Manage web traffic using the Azure CLI
	Restrict web traffic using the Azure CLI
app-service	Create an App Service app with continuous deployment from an Azure DevOps repository using Azure CLI
	Create an ASP.NET Core app in a Docker container in App Service from Azure Container Registry
azure-app-configuration	Create an Azure App Configuration store with the Azure CLI
	Delete an Azure App Configuration store with the Azure CLI
	Export from an Azure App Configuration store

Subject area	Sample name
	Import to an Azure App Configuration store
	Work with key-values in an Azure App Configuration store
azure-cache-for-redis	Create a Premium Azure Cache for Redis with clustering
	Create an Azure Cache for Redis using the Azure CLI
	Quickstart: Route Azure Cache for Redis events to web endpoint with Azure CLI
azure-devops	Use a variable group's secret and nonsecret variables in an Azure Pipeline
azure-functions	Create a function app for serverless code execution
	Create a function app in a Premium plan - Azure CLI
	Create a Function App in an App Service plan
	Create a function app in Azure that is deployed from GitHub
	Create a function app with a named Storage account connection
	Create a serverless Python function app using Azure CLI
	Create an Azure Function that connects to an Azure Cosmos DB
	Mount a file share to a Python function app using Azure CLI
azure-monitor	Create metric alert in Azure CLI
	Managing Azure Monitor Logs in Azure CLI
azure-signalr	Create a SignalR Service
	Create a SignalR Service with an App Service
	Create a web app that uses SignalR Service and GitHub authentication
azure-sql-edge	Install software and set up resources for the tutorial
batch	CLI example: Add an application to an Azure Batch account
	CLI example: Create a Batch account in Batch service mode
	CLI example: Create a Batch account in user subscription mode
	CLI example: Create and manage a Linux pool in Azure Batch

Subject area	Sample name
	CLI example: Create and manage a Windows pool in Azure Batch
	CLI example: Run a job and tasks with Azure Batch
cdn	Create an Azure Content Delivery Network profile and endpoint using the Azure CLI
chaos-studio	Use REST APIs to interact with Chaos Studio
cosmos-db	Add regions, change failover priority, trigger failover for an Azure Cosmos DB account using Azure CLI
	Connect an existing Azure Cosmos DB account with virtual network service endpoints using Azure CLI
	Create a database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Create a database with autoscale and shared collections for API for MongoDB for Azure Cosmos DB using Azure CLI
	Create a resource lock for Azure Cosmos DB Cassandra API keyspace and table using Azure CLI
	Create a resource lock for Azure Cosmos DB for Gremlin database and graph using Azure CLI
	Create a resource lock for Azure Cosmos DB's API for MongoDB using Azure CLI
	Create a serverless database and collection for API for MongoDB for Azure Cosmos DB using Azure CLI
	Create an Azure Cosmos DB account with IP firewall using Azure CLI
	Create an Azure Cosmos DB account with virtual network service endpoints using Azure CLI
	Create an Azure Cosmos DB Cassandra API account, keyspace and table using Azure CLI
	Create an Azure Cosmos DB Cassandra API serverless account, keyspace and table using Azure CLI
	Create an Azure Cosmos DB for Gremlin account, database and graph using Azure CLI
	Create an Azure Cosmos DB for NoSQL account, database and container using Azure CLI

Subject area	Sample name
	Create an Azure Cosmos DB for NoSQL account, database, and container with autoscale
	Create an Azure Cosmos DB for NoSQL serverless account, database and container using Azure CLI
	Create an Azure Cosmos DB Table API account and table using Azure CLI
	Create resource lock for an Azure Cosmos DB for NoSQL database and container using Azure CLI
	Find an existing Azure Cosmos DB free-tier account in a subscription using Azure CLI
	How to use the Azure SDK for Go with Azure Table
	Quickstart: Build an API for Table app with Python SDK and Azure Cosmos DB
	Throughput (RU/s) operations with Azure CLI for a database or container for Azure Cosmos DB for NoSQL
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB - API for Gremlin
	Throughput (RU/s) operations with Azure CLI for a database or graph for Azure Cosmos DB for MongoDB
	Throughput (RU/s) operations with Azure CLI for a keyspace or table for Azure Cosmos DB - API for Cassandra
	Throughput (RU/s) operations with Azure CLI for a table for Azure Cosmos DB for Table
	Use Azure CLI for resource lock operations on Azure Cosmos DB for Table tables
	Use Azure CLI to create a API for Cassandra account, keyspace, and table with autoscale
	Use Azure CLI to create a API for Gremlin account, database, and graph with autoscale
	Use Azure CLI to create a Gremlin serverless account, database, and graph
	Use Azure CLI to create an Azure Cosmos DB for Table account and table with autoscale

Subject area	Sample name
	Use Azure CLI to create an Azure Cosmos DB for Table serverless account and table
	Work with account keys and connection strings for an Azure Cosmos DB account using Azure CLI
defender-for-cloud	Azure Resource Graph sample queries for Microsoft Defender for Cloud
	Express configuration Azure Command Line Interface (CLI) commands reference
devtest-labs	Azure CLI Samples for Azure DevTest Labs
dns	Azure CLI script example: Create a DNS zone and record
event-grid	Create custom topic and subscribe to events for an Azure subscription with Azure CLI
frontdoor	Azure Front Door: Deploy custom domain
governance	Advanced Resource Graph query samples
	Azure Resource Graph sample queries by category
	Azure Resource Graph sample queries by table
	Azure Resource Graph sample queries for Azure Policy
	Azure Resource Graph sample queries for management groups
	Starter Resource Graph query samples
hdinsight	Azure HDInsight: Azure CLI samples
logic-apps	Azure CLI script sample - create a logic app
mariadb	Create a MariaDB server and configure a firewall rule using the Azure CLI
	Create a MariaDB server and configure a vNet rule using the Azure CLI
	Enable and download server slow query logs of an Azure Database for MariaDB server using Azure CLI
	List and update configurations of an Azure Database for MariaDB server using Azure CLI
	Monitor and scale an Azure Database for MariaDB server using Azure CLI

Subject area	Sample name
	Restore an Azure Database for MariaDB server using Azure CLI
mysql	Configure audit logs on Azure Database for MySQL - Flexible Server using Azure CLI
	Configure same-zone high availability in Azure Database for MySQL - Flexible Server using Azure CLI
	Configure slow query logs on an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Configure zone-redundant high availability in an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Create a MySQL server and configure a firewall rule using the Azure CLI
	Create an Azure Database for MySQL - Flexible Server database in a VNet using Azure CLI
	Create an Azure Database for MySQL - Flexible Server instance and enable public access connectivity using Azure CLI
	Create and manage read replicas in Azure Database for MySQL - Flexible Server using Azure CLI
	Enable and download server slow query logs of an Azure Database for MySQL server using Azure CLI
	List and change server parameters of Azure Database for MySQL - Flexible Server using Azure CLI
	List and update configurations of an Azure Database for MySQL server using Azure CLI
	Monitor and scale an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Monitor and scale an Azure Database for MySQL server using Azure CLI
	Restart/stop/start an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Restore an Azure Database for MySQL - Flexible Server instance using Azure CLI
	Restore an Azure Database for MySQL server using Azure CLI
networking	Load balance multiple websites

Subject area	Sample name
	Route traffic for high availability of applications - Azure CLI
postgresql	Create an Azure Database for PostgreSQL - Flexible Server instance and configure a firewall rule using the Azure CLI
	Create an Azure Database for PostgreSQL - Flexible Server instance and configure a vNet rule using the Azure CLI
	Enable and download server slow query logs of an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
	List and update configurations of an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
	Monitor and scale a single Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
	Restore an Azure Database for PostgreSQL - Flexible Server instance using Azure CLI
service-fabric	Create a secure Service Fabric Linux cluster via the Azure CLI
service-health	Azure Resource Graph sample queries for Azure Service Health
sql-server	Add a database to a failover group using the Azure CLI
	Add an Azure SQL Database elastic pool to a failover group using the Azure CLI
	Azure CLI script to enable transparent data encryption using your own key
	Backup an Azure SQL single database to an Azure storage container using the Azure CLI
	Configure a failover group for a group of databases in Azure SQL Database using the Azure CLI
	Configure active geo-replication for a pooled database in Azure SQL Database using the Azure CLI
	Configure active geo-replication for a single database in Azure SQL Database using the Azure CLI
	Copy a database in Azure SQL Database to a new server using the Azure CLI
	Create a single database and configure a firewall rule using the Azure CLI

Subject area	Sample name
	Create an Azure SQL Managed Instance using the Azure CLI
	Import a BACPAC file into a database in SQL Database using the Azure CLI
	Monitor and scale a single database in Azure SQL Database using the Azure CLI
	Move a database in SQL Database in a SQL elastic pool using the Azure CLI
	Restore a single database in Azure SQL Database to an earlier point in time using the Azure CLI
	Scale an elastic pool in Azure SQL Database using the Azure CLI
storage	Calculate the size of a Blob storage container
	Create a storage account and rotate its account access keys
	Use an Azure CLI script to delete containers based on container name prefix
virtual-machines	Azure Resource Graph sample queries for Azure Virtual Machines
	Copy managed disks to same or different subscription with CLI
	Copy snapshot of a managed disk to same or different subscription with CLI
	Create a managed disk from a snapshot with CLI (Linux)
	Create a managed disk from a VHD file in a storage account in the same subscription with CLI (Linux)
	Create a virtual machine from a snapshot with CLI
	Create a virtual machine using an existing managed OS disk with CLI
	Export/Copy a managed disk to a storage account using the Azure CLI
	Export/Copy a snapshot to a storage account in different region with CLI
	Move a Marketplace Azure Virtual Machine to another subscription
virtual-network	Peer two virtual networks with an Azure CLI script sample

Azure CLI samples for Azure Blob storage

Article • 11/18/2022

The following table includes links to Bash scripts built using the Azure CLI that create and manage Azure Storage.

Script	Description
Storage accounts	
Create a storage account and retrieve/rotate the access keys	Creates an Azure Storage account and retrieves and rotates its access keys.
Blob storage	
Calculate the total size of a Blob storage container	Calculates the total size of all the blobs in a container.
Delete containers with a specific prefix	Deletes containers starting with a specified string.

Manage Azure Cache for Redis with Azure CLI

Article • 03/24/2022

The following table includes links to bash scripts built using the Azure CLI.

Create cache	Description
Create and manage a cache	Creates a resource group and a basic tier Azure Cache for Redis. It then gets details of an Azure Cache for Redis instance, including provisioning status, the hostname, ports, and keys for an Azure Cache for Redis instance. Finally, it deletes the cache.
Create and manage a premium cache with clustering	Creates a resource group and a premium tier cache with clustering enabled. It then gets details of an Azure Cache for Redis instance, including provisioning status, the hostname, ports, and keys for an Azure Cache for Redis instance. Finally, it deletes the cache.

For more information about the Azure CLI, see [Install the Azure CLI](#) and [Get started with Azure CLI](#).

Azure CLI samples for Azure Cosmos DB for NoSQL

Article • 08/14/2024

APPLIES TO:  NoSQL

The following tables include links to sample Azure CLI scripts for the Azure Cosmos DB for NoSQL and to sample Azure CLI scripts that apply to all Azure Cosmos DB APIs. Common samples are the same across all APIs.

These samples require Azure CLI version 2.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#). If using Azure Cloud Shell, the latest version is already installed.

API for NoSQL Samples

 [Expand table](#)

Task	Description
Create an Azure Cosmos DB account, database, and container	Creates an Azure Cosmos DB account, database, and container for API for NoSQL.
Create a serverless Azure Cosmos DB account, database, and container	Creates a serverless Azure Cosmos DB account, database, and container for API for NoSQL.
Create an Azure Cosmos DB account, database, and container with autoscale	Creates an Azure Cosmos DB account, database, and container with autoscale for API for NoSQL.
Perform throughput operations	Read, update, and migrate between autoscale and standard throughput on a database and container.
Lock resources from deletion	Prevent resources from being deleted with resource locks.

Common API Samples

These samples apply to all Azure Cosmos DB APIs. These samples use a API for NoSQL account, but these operations are identical across all database APIs in Azure Cosmos DB.

 [Expand table](#)

Task	Description
Add or fail over regions	Add a region, change failover priority, trigger a manual failover.
Perform account key operations	List account keys, read-only keys, regenerate keys and list connection strings.
Secure with IP firewall	Create an Azure Cosmos DB account with IP firewall configured.
Secure new account with service endpoints	Create an Azure Cosmos DB account and secure with service-endpoints.
Secure existing account with service endpoints	Update an Azure Cosmos DB account to secure with service-endpoints when the subnet is eventually configured.
Find existing free-tier account	Find whether there is an existing free-tier account in your subscription.

Next steps

Reference pages for all Azure Cosmos DB CLI commands are available in the [Azure CLI Reference](#).

For Azure CLI samples for other APIs see:

- [CLI Samples for Cassandra](#)
- [CLI Samples for Gremlin](#)
- [CLI Samples for API for MongoDB](#)
- [CLI Samples for Table](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Azure CLI samples for Azure Database for MariaDB

Article • 08/14/2024

ⓘ Important

Azure Database for MariaDB is on the retirement path. We strongly recommend that you migrate to Azure Database for MySQL. For more information about migrating to Azure Database for MySQL, see [What's happening to Azure Database for MariaDB? ↗](#).

You can configure Azure SQL Database for MariaDB by using the [Azure CLI](#).

If you don't have an [Azure subscription](#), create an [Azure free account ↗](#) before you begin.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
A blue rectangular button with a white 'A' icon and the text 'Launch Cloud Shell' next to it, followed by a small blue square with a white arrow pointing right.
A blue rectangular button with a white 'A' icon and the text 'Launch Cloud Shell' next to it, followed by a small blue square with a white arrow pointing right.
- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

Samples

The following table includes links to sample Azure CLI scripts for Azure Database for MariaDB.

[Expand table](#)

Sample link	Description
Create a server with firewall rule	
Create a server and firewall rule	Azure CLI script that creates a single Azure Database for MariaDB server and configures a server-level firewall rule.
Create script with vNet rules	Azure CLI that creates an Azure Database for MariaDB server with a service endpoint on a virtual network and configures a vNet rule.
Scale a server	
Scale a server	Azure CLI script that scales a single Azure Database for MariaDB server up or down to allow for changing performance needs.
Change server configurations	
Change server configurations	Azure CLI script that change configurations of a single Azure Database for MariaDB server.
Restore a server	
Restore a server	Azure CLI script that restores a single Azure Database for MariaDB server to a previous point in time.
Manipulate with server logs	
Enable server logs	Azure CLI script that enables server logs of a single Azure Database for MariaDB server.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Azure CLI samples for Azure Database for MySQL - Flexible Server

Article • 08/14/2024

APPLIES TO:  Azure Database for MySQL - Flexible Server

The following table includes links to sample Azure CLI scripts for Azure Database for MySQL flexible server.

 Expand table

Sample link	Description
Create and connect to a server	
Create a server and enable public access connectivity	Creates an Azure Database for MySQL flexible server instance, configures a server-level firewall rule (public access connectivity method) and connects to the server.
Create a server and enable private access connectivity (VNet Integration)	Creates an Azure Database for MySQL flexible server instance in a VNet (private access connectivity method) and connects to the server through a VM within the VNet.
Monitor and scale	
Monitor metrics and scale a server	Monitors and scales a single Azure Database for MySQL flexible server instance up or down to allow for changing performance needs.
Backup and restore	
Restore a server	Restores a single Azure Database for MySQL flexible server instance to a previous point in time.
High Availability	
Configure zone-redundant high availability	Enables Zone-Redundant high availability while creating an Azure Database for MySQL flexible server instance.
Configure same-zone high availability	Enables Same-Zone high availability while creating an Azure Database for MySQL flexible server instance.
Manage server	
Restart, Stop, Start a Server	Performs restart, stop and start operations on a single Azure Database for MySQL flexible server instance.

Sample link	Description
Change server parameters	Changes server parameters of a single Azure Database for MySQL flexible server instance.
Replication	
Create read replicas	Creates and manages read replicas in a single Azure Database for MySQL flexible server instance.
Configure logs	
Configure audit logs	Configures audit logs on a single Azure Database for MySQL flexible server instance.
Configure slow-query logs	Configures slow-query logs on a single Azure Database for MySQL flexible server instance.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Get help at Microsoft Q&A](#)

Azure CLI samples for Azure SQL Database

Article • 04/30/2024

Applies to:  Azure SQL Database

You can configure Azure SQL Database by using the [Azure CLI](#).

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).

- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

Samples

The following table includes links to Azure CLI script examples to manage single and pooled databases in Azure SQL Database.

 Expand table

Area	Description
Create databases	
Create a single database	Creates an SQL Database and configures a server-level firewall rule.
Create pooled databases	Creates elastic pools, moves pooled databases, and changes compute sizes.
Scale databases	
Scale a single database	Scales single database.
Scale pooled database	Scales a SQL elastic pool to a different compute size.
Configure geo-replication	
Single database	Configures active geo-replication for a database in Azure SQL Database and fails it over to the secondary replica.
Pooled database	Configures active geo-replication for a database in an elastic pool, then fails it over to the secondary replica.
Configure failover group	
Configure failover group	Configures a failover group for a group of databases and failover over databases to the secondary server.
Single database	Creates a database and a failover group, adds the database to the failover group, then tests failover to the secondary server.
Pooled database	Creates a database, adds it to an elastic pool, adds the elastic pool to the failover group, then tests failover to the secondary server.
Back up, restore, copy, and import a database	
Back up a database	Backs up a database in SQL Database to an Azure storage backup.
Restore a database	Restores a database in SQL Database to a specific point in time.
Copy a database to a new server	Creates a copy of an existing database in SQL Database in a new server.
Import a database from a BACPAC file	Imports a database to SQL Database from a BACPAC file.

Learn more about the [single-database Azure CLI API](#).

Related content

- Az sql cmdlets
 - PowerShell scripts
-

Feedback

Was this page helpful?

 Yes

 No

Provide product feedback 

Azure CLI Samples

Article • 04/22/2024

These end-to-end Azure CLI scripts are provided to help you learn how to provision and managing the Azure resources required by Azure Functions. You must use the [Azure Functions Core Tools](#) to create actual Azure Functions code projects from the command line on your local computer and deploy code to these Azure resources. For a complete end-to-end example of developing and deploying from the command line using both Core Tools and the Azure CLI, see one of these language-specific command line quickstarts:

- [C#](#)
- [Java](#)
- [JavaScript](#)
- [PowerShell](#)
- [Python](#)
- [TypeScript](#)

The following table includes links to bash scripts that you can use to create and manage the Azure resources required by Azure Functions using the Azure CLI.

[+] Expand table

Create app	Description
Create a function app for serverless execution	Create a function app in a Consumption plan.
Create a serverless Python function app	Create a Python function app in a Consumption plan.
Create a function app in a scalable Premium plan	Create a function app in a Premium plan.
Create a function app in a dedicated (App Service) plan	Create a function app in a dedicated App Service plan.

[+] Expand table

Integrate	Description
Create a function app and connect to a storage account	Create a function app and connect it to a storage account.

Integrate	Description
Create a function app and connect to an Azure Cosmos DB	Create a function app and connect it to an Azure Cosmos DB instance.
Create a Python function app and mount an Azure Files share	By mounting a share to your Linux function app, you can leverage existing machine learning models or other data in your functions.

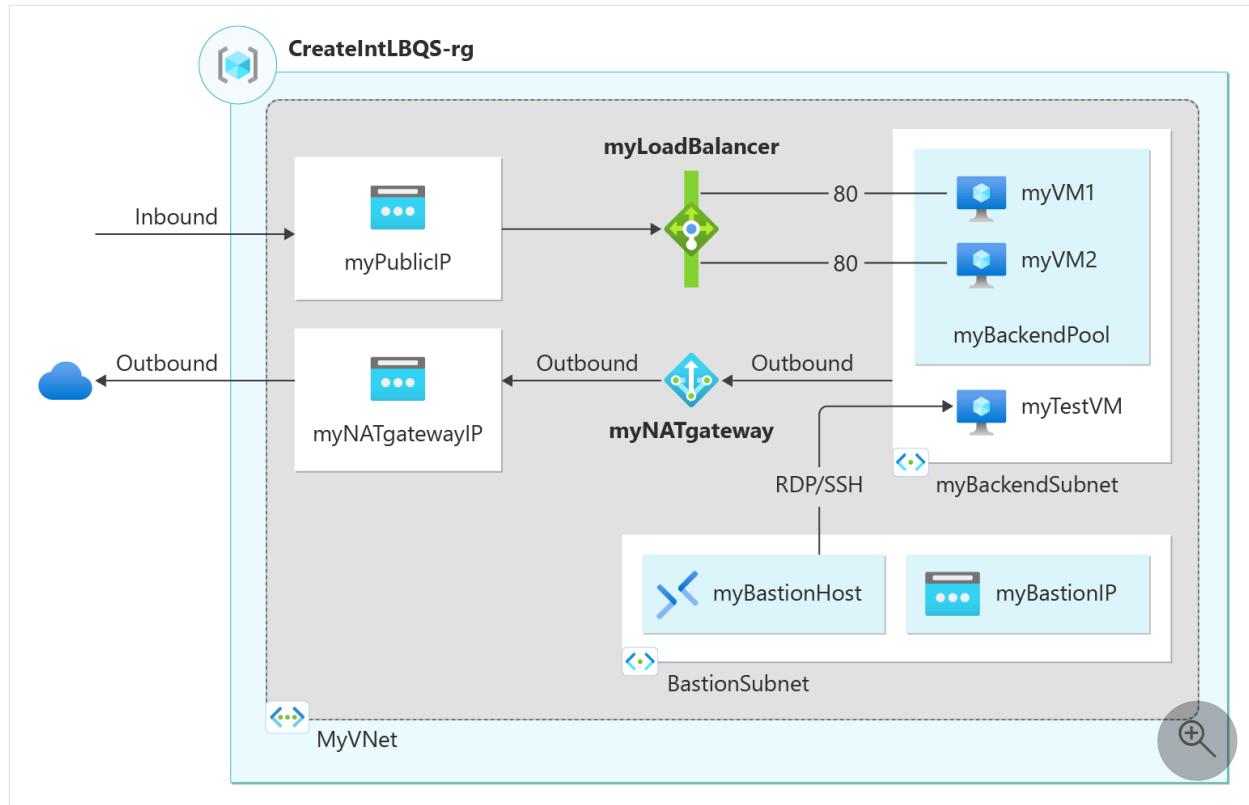
[\[\] Expand table](#)

Continuous deployment	Description
Deploy from GitHub	Create a function app that deploys from a GitHub repository.

Quickstart: Create a public load balancer to load balance VMs using the Azure CLI

Article • 09/30/2024

Get started with Azure Load Balancer by using the Azure CLI to create a public load balancer and two virtual machines. Along with these resources, you deploy Azure Bastion, NAT Gateway, a virtual network, and the required subnets.



If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).
- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).

[Launch Cloud Shell](#) [↗](#)

- If you're using a local installation, sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).
- This quickstart requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [az group create](#):

Azure CLI

```
az group create \
  --name CreatePubLBQS-rg \
  --location eastus
```

Create a virtual network

Before you deploy VMs and test your load balancer, create the supporting virtual network and subnet.

Create a virtual network using [az network vnet create](#). The virtual network and subnet contain the resources deployed later in this article.

Azure CLI

```
az network vnet create \
  --resource-group CreatePubLBQS-rg \
  --location eastus \
  --name myVNet \
  --address-prefixes 10.1.0.0/16 \
  --subnet-name myBackendSubnet \
  --subnet-prefixes 10.1.0.0/24
```

Create a public IP address

To access your web app on the Internet, you need a public IP address for the load balancer.

Use [az network public-ip create](#) to create the public IP for the load balancer frontend.

Azure CLI

```
az network public-ip create \
--resource-group CreatePubLBQS-rg \
--name myPublicIP \
--sku Standard \
--zone 1 2 3
```

To create a zonal public IP address in Zone 1 instead, use the following command:

Azure CLI

```
az network public-ip create \
--resource-group CreatePubLBQS-rg \
--name myPublicIP \
--sku Standard \
--zone 1
```

Create a load balancer

This section details how you can create and configure the following components of the load balancer:

- A frontend IP pool that receives the incoming network traffic on the load balancer
- A backend IP pool where the frontend pool sends the load balanced network traffic
- A health probe that determines health of the backend VM instances
- A load balancer rule that defines how traffic is distributed to the VMs

Create the load balancer resource

Create a public load balancer with [az network lb create](#):

Azure CLI

```
az network lb create \
--resource-group CreatePubLBQS-rg \
--name myLoadBalancer \
--sku Standard \
--public-ip-address myPublicIP \
--frontend-ip-name myFrontEnd \
--backend-pool-name myBackEndPool
```

If the public IP created is zonal, the specified zone needs to be defined when creating the public load balancer.

Azure CLI

```
az network lb create \
--resource-group CreatePubLBQS-rg \
--name myLoadBalancer \
--sku Standard \
--public-ip-address myPublicIP \
--frontend-ip-name myFrontEnd \
--public-ip-zone 1 \
--backend-pool-name myBackEndPool
```

Create the health probe

A health probe checks all virtual machine instances to ensure they can send network traffic.

A virtual machine with a failed probe check is removed from the load balancer. The virtual machine is added back into the load balancer when the failure is resolved.

Create a health probe with [az network lb probe create](#):

Azure CLI

```
az network lb probe create \
--resource-group CreatePubLBQS-rg \
--lb-name myLoadBalancer \
--name myHealthProbe \
--protocol tcp \
--port 80
```

Create the load balancer rule

A load balancer rule defines:

- Frontend IP configuration for the incoming traffic
- The backend IP pool to receive the traffic
- The required source and destination port

Create a load balancer rule with [az network lb rule create](#):

Azure CLI

```
az network lb rule create \
--resource-group CreatePubLBQS-rg \
--lb-name myLoadBalancer \
--name myHTTPRule \
--protocol tcp \
--frontend-port 80 \
--backend-port 80 \
--frontend-ip-name myFrontEnd \
--backend-pool-name myBackEndPool \
--probe-name myHealthProbe \
--disable-outbound-snat true \
--idle-timeout 15 \
--enable-tcp-reset true
```

Create a network security group

For a standard load balancer, the VMs in the backend pool are required to have network interfaces that belong to a network security group.

Use [az network nsg create](#) to create the network security group:

Azure CLI

```
az network nsg create \
--resource-group CreatePubLBQS-rg \
--name myNSG
```

Create a network security group rule

Create a network security group rule using [az network nsg rule create](#):

Azure CLI

```
az network nsg rule create \
--resource-group CreatePubLBQS-rg \
--nsg-name myNSG \
--name myNSGRuleHTTP \
```

```
--protocol '*' \
--direction inbound \
--source-address-prefix '*' \
--source-port-range '*' \
--destination-address-prefix '*' \
--destination-port-range 80 \
--access allow \
--priority 200
```

Create a bastion host

In this section, you create the resources for Azure Bastion. Azure Bastion is used to securely manage the virtual machines in the backend pool of the load balancer.

ⓘ Important

Hourly pricing starts from the moment that Bastion is deployed, regardless of outbound data usage. For more information, see [Pricing](#) and [SKUs](#). If you're deploying Bastion as part of a tutorial or test, we recommend that you delete this resource after you finish using it.

Create a public IP address

Use [az network public-ip create](#) to create a public ip address for the bastion host. The public IP is used by the bastion host for secure access to the virtual machine resources.

Azure CLI

```
az network public-ip create \
--resource-group CreatePubLBQS-rg \
--name myBastionIP \
--sku Standard \
--zone 1 2 3
```

Create a bastion subnet

Use [az network vnet subnet create](#) to create a bastion subnet. The bastion subnet is used by the bastion host to access the virtual network.

Azure CLI

```
az network vnet subnet create \
--resource-group CreatePubLBQS-rg \
```

```
--name AzureBastionSubnet \
--vnet-name myVNet \
--address-prefixes 10.1.1.0/27
```

Create bastion host

Use [az network bastion create](#) to create a bastion host. The bastion host is used to connect securely to the virtual machine resources created later in this article.

Azure CLI

```
az network bastion create \
--resource-group CreatePubLBQS-rg \
--name myBastionHost \
--public-ip-address myBastionIP \
--vnet-name myVNet \
--location eastus
```

It can take a few minutes for the Azure Bastion host to deploy.

Create backend servers

In this section, you create:

- Two network interfaces for the virtual machines
- Two virtual machines to be used as backend servers for the load balancer

Create network interfaces for the virtual machines

Create two network interfaces with [az network nic create](#):

Azure CLI

```
array=(myNicVM1 myNicVM2)
for vmnic in "${array[@]}"
do
    az network nic create \
        --resource-group CreatePubLBQS-rg \
        --name $vmnic \
        --vnet-name myVNet \
        --subnet myBackEndSubnet \
        --network-security-group myNSG
done
```

Create virtual machines

Create the virtual machines with [az vm create](#):

Azure CLI

```
az vm create \
--resource-group CreatePubLBQS-rg \
--name myVM1 \
--nics myNicVM1 \
--image win2019datacenter \
--admin-username azureuser \
--zone 1 \
--no-wait
```

Azure CLI

```
az vm create \
--resource-group CreatePubLBQS-rg \
--name myVM2 \
--nics myNicVM2 \
--image win2019datacenter \
--admin-username azureuser \
--zone 2 \
--no-wait
```

It may take a few minutes for the VMs to deploy. You can continue to the next steps while the VMs are creating.

ⓘ Note

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the backend pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

The default outbound access IP is disabled when one of the following events happens:

- A public IP address is assigned to the VM.
- The VM is placed in the backend pool of a standard load balancer, with or without outbound rules.
- An [Azure NAT Gateway](#) resource is assigned to the subnet of the VM.

VMs that you create by using virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Default outbound access in Azure](#) and [Use Source Network Address Translation \(SNAT\) for outbound connections](#).

Add virtual machines to load balancer backend pool

Add the virtual machines to the backend pool with `az network nic ip-config address-pool add`:

Azure CLI

```
array=(myNicVM1 myNicVM2)
for vmnic in "${array[@]}"
do
    az network nic ip-config address-pool add \
        --address-pool myBackendPool \
        --ip-config-name ipconfig1 \
        --nic-name $vmnic \
        --resource-group CreatePubLBQS-rg \
        --lb-name myLoadBalancer
done
```

Create NAT gateway

To provide outbound internet access for resources in the backend pool, create a NAT gateway.

Create public IP

Use `az network public-ip create` to create a single IP for the outbound connectivity.

Azure CLI

```
az network public-ip create \
    --resource-group CreatePubLBQS-rg \
    --name myNATgatewayIP \
    --sku Standard \
    --zone 1 2 3
```

To create a zonal redundant public IP address in Zone 1 instead, use the following command:

```
Azure CLI

az network public-ip create \
--resource-group CreatePubLBQS-rg \
--name myNATgatewayIP \
--sku Standard \
--zone 1
```

Create NAT gateway resource

Use [az network nat gateway create](#) to create the NAT gateway resource. The public IP created in the previous step is associated with the NAT gateway.

```
Azure CLI

az network nat gateway create \
--resource-group CreatePubLBQS-rg \
--name myNATgateway \
--public-ip-addresses myNATgatewayIP \
--idle-timeout 10
```

Associate NAT gateway with subnet

Configure the source subnet in virtual network to use a specific NAT gateway resource with [az network vnet subnet update](#).

```
Azure CLI

az network vnet subnet update \
--resource-group CreatePubLBQS-rg \
--vnet-name myVNet \
--name myBackendSubnet \
--nat-gateway myNATgateway
```

Install IIS

Use [az vm extension set](#) to install IIS on the virtual machines and set the default website to the computer name.

```
Azure CLI
```

```
array=(myVM1 myVM2)
for vm in "${array[@]}"
do
    az vm extension set \
        --publisher Microsoft.Compute \
        --version 1.8 \
        --name CustomScriptExtension \
        --vm-name $vm \
        --resource-group CreatePubLBQS-rg \
        --settings '{"commandToExecute":"powershell Add-WindowsFeature Web-Server; powershell Add-Content -Path \"C:\\inetpub\\wwwroot\\Default.htm\" -Value $($env:computername)"}'
done
```

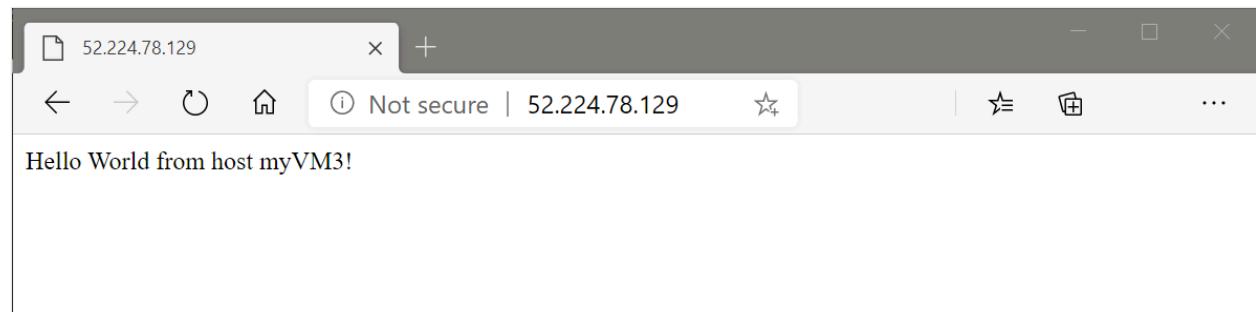
Test the load balancer

To get the public IP address of the load balancer, use [az network public-ip show](#).

Copy the public IP address, and then paste it into the address bar of your browser.

Azure CLI

```
az network public-ip show \
    --resource-group CreatePubLBQS-rg \
    --name myPublicIP \
    --query ipAddress \
    --output tsv
```



Clean up resources

When no longer needed, use the [az group delete](#) command to remove the resource group, load balancer, and all related resources.

Azure CLI

```
az group delete \
    --name CreatePubLBQS-rg
```

Next steps

In this quickstart:

- You created a standard public load balancer
- Attached two virtual machines
- Configured the load balancer traffic rule and health probe
- Tested the load balancer

To learn more about Azure Load Balancer, continue to:

[What is Azure Load Balancer?](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Quickstart: Create and publish an Azure Managed Application definition

Article • 09/22/2024

This quickstart provides an introduction to working with [Azure Managed Applications](#). You create and publish a managed application definition that's stored in your service catalog and is intended for members of your organization.

To publish a managed application to your service catalog, do the following tasks:

- Create an Azure Resource Manager template (ARM template) that defines the resources that deploy with the managed application.
- Define the user interface elements for the portal when deploying the managed application.
- Create a *.zip* package that contains the required JSON files. The *.zip* package file has a 120-MB limit for a service catalog's managed application definition.
- Publish the managed application definition so it's available in your service catalog.

If your managed application definition is more than 120 MB or if you want to use your own storage account for your organization's compliance reasons, go to [Quickstart: Bring your own storage to create and publish an Azure Managed Application definition](#).

You can use Bicep to develop a managed application definition but it must be converted to ARM template JSON before you can publish the definition in Azure. For more information, go to [Quickstart: Use Bicep to create and publish an Azure Managed Application definition](#).

You can also use Bicep to deploy a managed application definition from your service catalog. For more information, go to [Quickstart: Use Bicep to deploy an Azure Managed Application definition](#).

Prerequisites

To complete this quickstart, you need the following items:

- An Azure account with an active subscription and permissions to Microsoft Entra resources like users, groups, or service principals. If you don't have an account, [create a free account](#) before you begin.
- [Visual Studio Code](#) with the latest [Azure Resource Manager Tools extension](#). For Bicep files, install the [Bicep extension for Visual Studio Code](#).
- Install the latest version of [Azure PowerShell](#) or [Azure CLI](#).

Create the ARM template

Every managed application definition includes a file named *mainTemplate.json*. The template defines the Azure resources to deploy and is no different than a regular ARM template.

Open Visual Studio Code, create a file with the case-sensitive name *mainTemplate.json* and save it.

Add the following JSON and save the file. It defines the resources to deploy an App Service and App Service plan. The template uses the App Service Basic plan (B1) that has pay-as-you-go costs. For more information, see [Azure App Service on Linux pricing](#).

```
JSON

{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "location": {
            "type": "string",
            "defaultValue": "[resourceGroup().location]"
        },
        "appServicePlanName": {
            "type": "string",
            "maxLength": 40,
            "metadata": {
                "description": "App Service plan name."
            }
        },
        "appServiceNamePrefix": {
            "type": "string",
            "maxLength": 47,
            "metadata": {
                "description": "App Service name prefix."
            }
        }
    },
    "variables": {
        "appServicePlanSku": "B1",
        "appServicePlanCapacity": 1,
        "appServiceName": "[format('{0}{1}', parameters('appServiceNamePrefix'), uniqueString(resourceGroup().id))]",
        "linuxFxVersion": "DOTNETCORE|8.0"
    },
    "resources": [
        {
            "type": "Microsoft.Web/serverfarms",
            "apiVersion": "2023-01-01",
            "name": "[parameters('appServicePlanName')]"
        }
    ]
}
```

```

    "location": "[parameters('location')]",
    "sku": {
        "name": "[variables('appServicePlanSku')]",
        "capacity": "[variables('appServicePlanCapacity')]"
    },
    "kind": "linux",
    "properties": {
        "zoneRedundant": false,
        "reserved": true
    }
},
{
    "type": "Microsoft.Web/sites",
    "apiVersion": "2023-01-01",
    "name": "[variables('appServiceName')]",
    "location": "[parameters('location')]",
    "properties": {
        "serverFarmId": "[resourceId('Microsoft.Web/serverfarms',
parameters('appServicePlanName'))]",
        "httpsOnly": true,
        "redundancyMode": "None",
        "siteConfig": {
            "linuxFxVersion": "[variables('linuxFxVersion')]",
            "minTlsVersion": "1.2",
            "ftpsState": "Disabled"
        }
    },
    "dependsOn": [
        "[resourceId('Microsoft.Web/serverfarms',
parameters('appServicePlanName'))]"
    ]
},
],
"outputs": {
    "appServicePlan": {
        "type": "string",
        "value": "[parameters('appServicePlanName')]"
    },
    "appServiceApp": {
        "type": "string",
        "value": "[reference(resourceId('Microsoft.Web/sites',
variables('appServiceName')), '2023-01-01').defaultHostName]"
    }
}
}

```

Define your portal experience

As a publisher, you define the portal experience to create the managed application. The *createUiDefinition.json* file generates the portal's user interface. You define how users

provide input for each parameter using **control elements** like drop-downs and text boxes.

In this example, the user interface prompts you to input the App Service name prefix and App Service plan's name. During deployment of `mainTemplate.json` the `appServiceName` variables uses the `uniqueString` function to append a 13-character string to the name prefix so the name is globally unique across Azure.

Open Visual Studio Code, create a file with the case-sensitive name *createUiDefinition.json* and save it.

Add the following JSON code to the file and save it.

JSON

```
{  
  "$schema": "https://schema.management.azure.com/schemas/0.1.2-  
preview/CreateUIDefinition.MultiVm.json#",  
  "handler": "Microsoft.Azure.CreateUIDef",  
  "version": "0.1.2-preview",  
  "parameters": {  
    "basics": [  
      {}  
    ],  
    "steps": [  
      {  
        "name": "webAppSettings",  
        "label": "Web App settings",  
        "subLabel": {  
          "preValidation": "Configure the web app settings",  
          "postValidation": "Completed"  
        },  
        "elements": [  
          {  
            "name": "appServicePlanName",  
            "type": "Microsoft.Common.TextBox",  
            "label": "App Service plan name",  
            "placeholder": "App Service plan name",  
            "defaultValue": "",  
            "toolTip": "Use alphanumeric characters or hyphens with a  
maximum of 40 characters.",  
            "constraints": {  
              "required": true,  
              "regex": "^[a-zA-Z0-9A-Z-]{1,40}$",  
              "validationMessage": "Only alphanumeric characters or hyphens  
are allowed, with a maximum of 40 characters."  
            },  
            "visible": true  
          },  
          {  
            "name": "appServiceName",  
            "type": "Microsoft.Common.TextBox",  
            "label": "App Service Name",  
            "placeholder": "App Service Name",  
            "defaultValue": "",  
            "toolTip": "The name of the App Service. It must be unique across all  
App Services in the same region.",  
            "constraints": {  
              "required": true,  
              "regex": "^[a-zA-Z0-9A-Z-]{1,40}$",  
              "validationMessage": "Only alphanumeric characters or hyphens  
are allowed, with a maximum of 40 characters."  
            },  
            "visible": true  
          }  
        ]  
      }  
    ]  
  }  
}
```

```
        "label": "App Service name prefix",
        "placeholder": "App Service name prefix",
        "defaultValue": "",
        "toolTip": "Use alphanumeric characters or hyphens with minimum of 2 characters and maximum of 47 characters.",
        "constraints": {
            "required": true,
            "regex": "^[a-zA-Z0-9-]{2,47}$",
            "validationMessage": "Only alphanumeric characters or hyphens are allowed, with a minimum of 2 characters and maximum of 47 characters."
        },
        "visible": true
    }
]
}
],
"outputs": {
    "location": "[location()]",
    "appServicePlanName": "[steps('webAppSettings').appServicePlanName]",
    "appServiceNamePrefix": "[steps('webAppSettings').appServiceName]"
}
}
}
```

To learn more, see [Get started with CreateUiDefinition](#).

Package the files

Add the two files to a package file named *app.zip*. The two files must be at the root level of the *.zip* file. If the files are in a folder, when you create the managed application definition, you receive an error that states the required files aren't present.

Upload *app.zip* to an Azure storage account so you can use it when you deploy the managed application's definition. The storage account name must be globally unique across Azure and the length must be 3-24 characters with only lowercase letters and numbers. In the command, replace the placeholder <pkgstorageaccountname> including the angle brackets (<>), with your unique storage account name.

Azure PowerShell

In Visual Studio Code, open a new PowerShell terminal and sign in to your Azure subscription.

Azure PowerShell

[Connect-AzAccount](#)

The command opens your default browser and prompts you to sign in to Azure. For more information, go to [Sign in with Azure PowerShell](#).

```
Azure PowerShell

New-AzResourceGroup -Name packageStorageGroup -Location westus

$pkgstorageparms = @{
    ResourceGroupName = "packageStorageGroup"
    Name = "<pkgstorageaccountname>"
    Location = "westus"
    SkuName = "Standard_LRS"
    Kind = "StorageV2"
    MinimumTlsVersion = "TLS1_2"
    AllowBlobPublicAccess = $true
    AllowSharedKeyAccess = $false
}

$pkgstorageaccount = New-AzStorageAccount @pkgstorageparms
```

The `$pkgstorageparms` variable uses PowerShell [splatting](#) to improve readability for the parameter values used in the command to create the new storage account. Splatting is used in other PowerShell commands that use multiple parameter values.

After you create the storage account, add the role assignment *Storage Blob Data Contributor* to the storage account scope. Assign access to your Microsoft Entra user account. Depending on your access level in Azure, you might need other permissions assigned by your administrator. For more information, see [Assign an Azure role for access to blob data](#) and [Assign Azure roles using the Azure portal](#).

After you add the role to the storage account, it takes a few minutes to become active in Azure. You can then create the context needed to create the container and upload the file.

```
Azure PowerShell

$pkgstoragecontext = New-AzStorageContext -StorageAccountName
$pkgstorageaccount.StorageAccountName -UseConnectedAccount

New-AzStorageContainer -Name appcontainer -Context $pkgstoragecontext -
Permission blob

$blobparms = @{
    File = "app.zip"
    Container = "appcontainer"
    Blob = "app.zip"
    Context = $pkgstoragecontext
}
```

```
Set-AzStorageBlobContent @blobparms
```

Create the managed application definition

In this section, you get identity information from Microsoft Entra ID, create a resource group, and deploy the managed application definition.

Get group ID and role definition ID

The next step is to select a user, security group, or application for managing the resources for the customer. This identity has permissions on the managed resource group according to the assigned role. The role can be any Azure built-in role like Owner or Contributor.

Azure PowerShell

This example uses a security group, and your Microsoft Entra account should be a member of the group. To get the group's object ID, replace the placeholder `<managedAppDemo>` including the angle brackets (`<>`), with your group's name. You use this variable's value when you deploy the managed application definition.

To create a new Microsoft Entra group, go to [Manage Microsoft Entra groups and group membership](#).

Azure PowerShell

```
$principalid=(Get-AzADGroup -DisplayName <managedAppDemo>).Id
```

Next, get the role definition ID of the Azure built-in role you want to grant access to the user, group, or application. You use this variable's value when you deploy the managed application definition.

Azure PowerShell

```
$roleid=(Get-AzRoleDefinition -Name Owner).Id
```

Publish the managed application definition

Azure PowerShell

Create a resource group for your managed application definition.

Azure PowerShell

```
New-AzResourceGroup -Name appDefinitionGroup -Location westus
```

The `blob` command creates a variable to store the URL for the package .zip file. That variable is used in the command that creates the managed application definition.

Azure PowerShell

```
$blob = Get-AzStorageBlob -Container appcontainer -Blob app.zip -Context $pkgstoragecontext

$publishparms = @{
    Name = "sampleManagedApplication"
    Location = "westus"
    ResourceGroupName = "appDefinitionGroup"
    LockLevel = "ReadOnly"
    DisplayName = "Sample managed application"
    Description = "Sample managed application that deploys web resources"
    Authorization = "${principalid}:$roleid"
    PackageFileUri = $blob.ICloudBlob.StorageUri.PrimaryUri.AbsoluteUri
}

New-AzManagedApplicationDefinition @publishparms
```

When the command completes, you have a managed application definition in your resource group.

Some of the parameters used in the preceding example are:

- `ResourceGroupName`: The name of the resource group where the managed application definition is created.
- `LockLevel`: The `lockLevel` on the managed resource group prevents the customer from performing undesirable operations on this resource group. Currently, `ReadOnly` is the only supported lock level. `ReadOnly` specifies that the customer can only read the resources present in the managed resource group. The publisher identities that are granted access to the managed resource group are exempt from the lock level.
- `Authorization`: Describes the principal ID and the role definition ID that are used to grant permission to the managed resource group.

- "\${principalid}:\$roleid" or you can use curly braces for each variable "\${principalid}:\${roleid}".
- Use a comma to separate multiple values: "\${principalid1}:\$roleid1", "\${principalid2}:\$roleid2".
- `PackageFileUri`: The location of a `.zip` package file that contains the required files.

Make sure users can see your definition

You have access to the managed application definition, but you want to make sure other users in your organization can access it. Grant them at least the Reader role on the definition. They might have inherited this level of access from the subscription or resource group. To check who has access to the definition and add users or groups, see [Assign Azure roles using the Azure portal](#).

Clean up resources

If you're going to deploy the definition, continue with the **Next steps** section that links to the article to deploy the definition.

If you're finished with the managed application definition, you can delete the resource groups you created named `packageStorageGroup` and `appDefinitionGroup`.

Azure PowerShell

The command prompts you to confirm that you want to remove the resource group.

Azure PowerShell

```
Remove-AzResourceGroup -Name packageStorageGroup  
Remove-AzResourceGroup -Name appDefinitionGroup
```

Next steps

You published the managed application definition. The next step is to learn how to deploy an instance of that definition.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Azure CLI samples for Azure Monitor

Article • 06/20/2023

Prerequisites

- Use the Bash environment in [Azure Cloud Shell](#). For more information, see [Quickstart for Bash in Azure Cloud Shell](#).

 [Launch Cloud Shell](#) 

- If you prefer to run CLI reference commands locally, [install](#) the Azure CLI. If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
 - If you're using a local installation, sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
 - When you're prompted, install the Azure CLI extension on first use. For more information about extensions, see [Use extensions with the Azure CLI](#).
 - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

Samples

The following table includes links to Azure CLI scripts written for a Bash environment that manage Azure Monitor.

Script	Description
Create metric alert monitors in Azure CLI	Create an alert for a virtual machine or create an alert that includes a dimension for an App Service Plan.
Managing Azure Monitor Logs in Azure CLI	Manage your log analytics workspace. Use these scripts to manage linked services, linked storage, intelligence packs, and saved searches.

Azure CLI reference

Article • 04/09/2023

The following table includes links to bash scripts for the Azure SignalR Service using the Azure CLI.

Script	Descriptions
Create	
Create a new SignalR Service and resource group	Creates a new Azure SignalR Service resource in a new resource group with a random name.
Integrate	
Create a new SignalR Service and Web App configured to use SignalR	Creates a new Azure SignalR Service resource in a new resource group with a random name. Also adds a new Web App and App Service plan to host an ASP.NET Core Web App that uses the SignalR Service. The web app is configured with an App Setting to connect to the new SignalR service resource.
Create a new SignalR Service and Web App configured to use SignalR, and GitHub OAuth	Creates a new Azure SignalR Service resource in a new resource group with a random name. Also adds a new Azure Web App and hosting plan to host an ASP.NET Core Web App that uses the SignalR Service. The web app is configured with app settings for the connection string to the new SignalR service resource, and client secrets to support GitHub authentication as demonstrated in the authentication tutorial . The web app is also configured to use a local git repository deployment source.

Azure networking services overview

Article • 08/28/2024

The networking services in Azure provide various networking capabilities that can be used together or separately. Select each of the following networking scenarios to learn more about them:

- **Networking foundation:** Azure networking foundation services provide core connectivity for your resources in Azure - Virtual Network (VNet), Private Link, Azure DNS, Azure Virtual Network Manager, Azure Bastion, Route Server, NAT Gateway, Traffic Manager, Azure Network Watcher, and Azure Monitor.
- **Load balancing and content delivery:** Azure load balancing and content delivery services allow for management, distribution, and optimization of your applications and workloads - Load balancer, Application Gateway, and Azure Front Door.
- **Hybrid connectivity:** Azure hybrid connectivity services secure communication to and from your resources in Azure - VPN Gateway, ExpressRoute, Virtual WAN, and Peering Service.
- **Network security:** Azure network security services protect your web applications and IaaS services from DDoS attacks and malicious actors - Firewall Manager, Firewall, Web Application Firewall, and DDoS Protection.

Networking foundation

This section describes services that provide the building blocks for designing and architecting a network environment in Azure - Virtual Network (VNet), Private Link, Azure DNS, Azure Virtual Network Manager, Azure Bastion, Route Server, NAT Gateway, Traffic Manager, Azure Network Watcher, and Azure Monitor.

Virtual network

[Azure Virtual Network \(VNet\)](#) is the fundamental building block for your private network in Azure. You can use VNets to:

- **Communicate between Azure resources:** You can deploy virtual machines, and several other types of Azure resources to a virtual network, such as Azure App Service Environments, the Azure Kubernetes Service (AKS), and Azure Virtual Machine Scale Sets. To view a complete list of Azure resources that you can deploy into a virtual network, see [Virtual network service integration](#).
- **Communicate between each other:** You can connect virtual networks to each other, enabling resources in either virtual network to communicate with each

other, using virtual network peering or Azure Virtual Network Manager. The virtual networks you connect can be in the same, or different, Azure regions. For more information, see [Virtual network peering](#) and [Azure Virtual Network Manager](#).

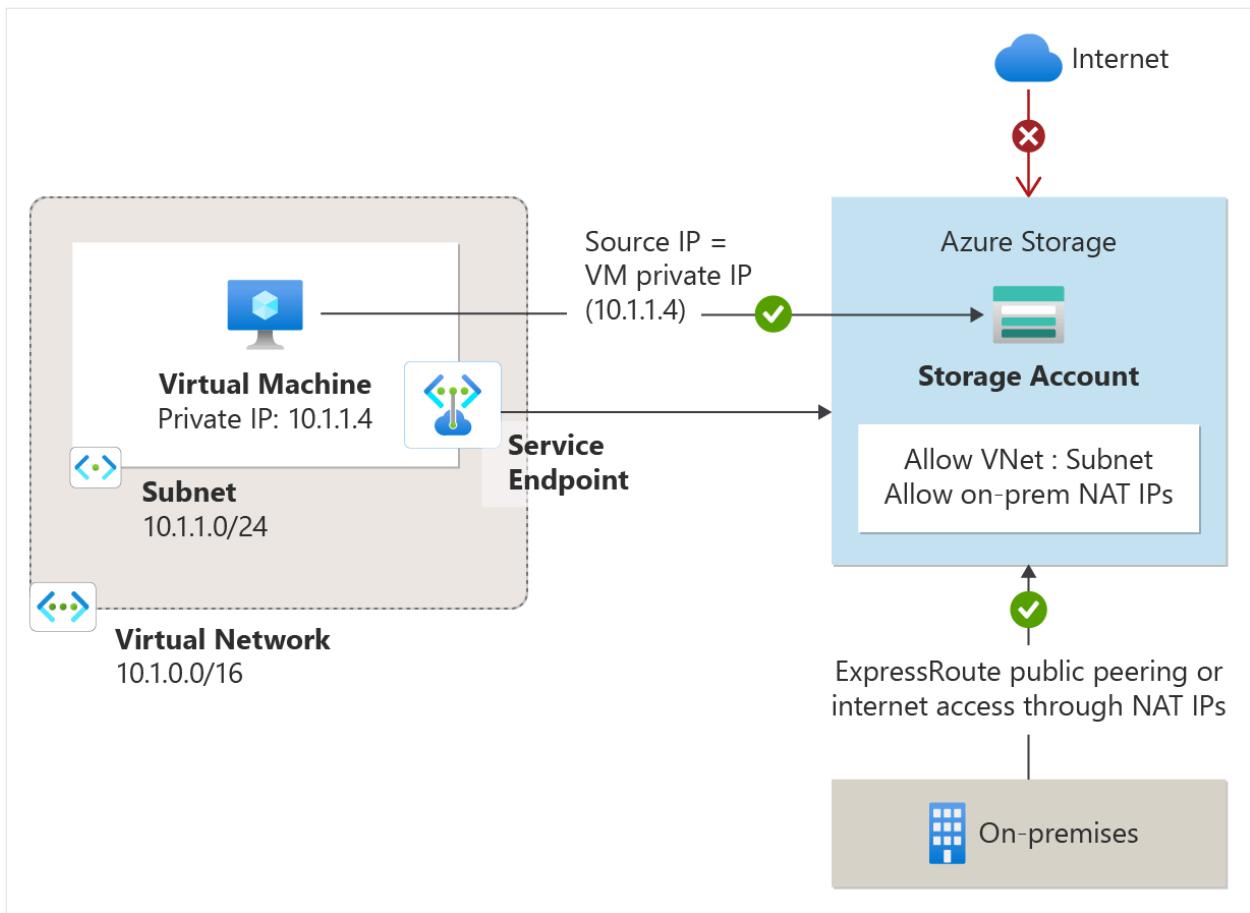
- **Communicate to the internet:** All resources in a virtual network can communicate outbound to the internet, by default. You can communicate inbound to a resource by assigning a public IP address or a public Load Balancer. You can also use [Public IP addresses](#) or public [Load Balancer](#) to manage your outbound connections.
- **Communicate with on-premises networks:** You can connect your on-premises computers and networks to a virtual network using [VPN Gateway](#) or [ExpressRoute](#).
- **Encrypt traffic between resources:** You can use [Virtual network encryption](#) to encrypt traffic between resources in a virtual network.

Network security groups

You can filter network traffic to and from Azure resources in an Azure virtual network with a network security group. For more information, see [Network security groups](#).

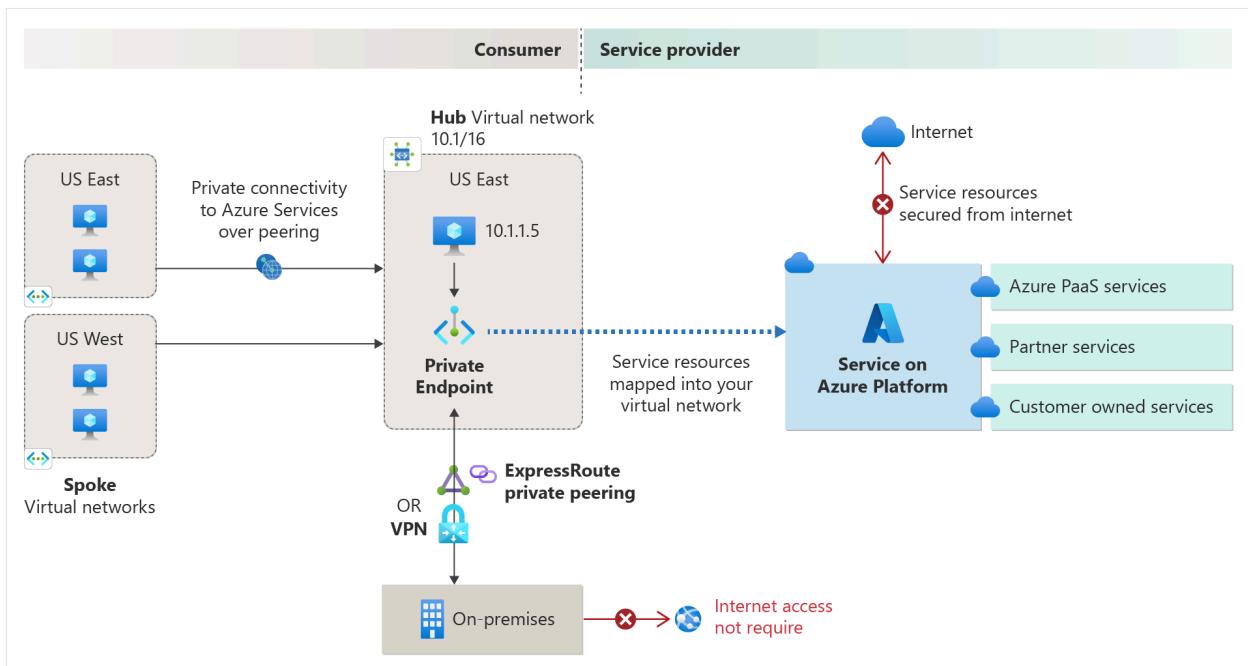
Service endpoints

[Virtual Network \(VNet\) service endpoints](#) extend your virtual network private address space and the identity of your virtual network to the Azure services, over a direct connection. Endpoints allow you to secure your critical Azure service resources to only your virtual networks. Traffic from your virtual network to the Azure service always remains on the Microsoft Azure backbone network.



Azure Private Link

[Azure Private Link](#) enables you to access Azure PaaS Services (for example, Azure Storage and SQL Database) and Azure hosted customer-owned/partner services over a private endpoint in your virtual network. Traffic between your virtual network and the service travels through the Microsoft backbone network. Exposing your service to the public internet is no longer necessary. You can create your own private link service in your virtual network and deliver it to your customers.



Azure DNS

Azure DNS provides DNS hosting and resolution using the Microsoft Azure infrastructure. Azure DNS consists of three services:

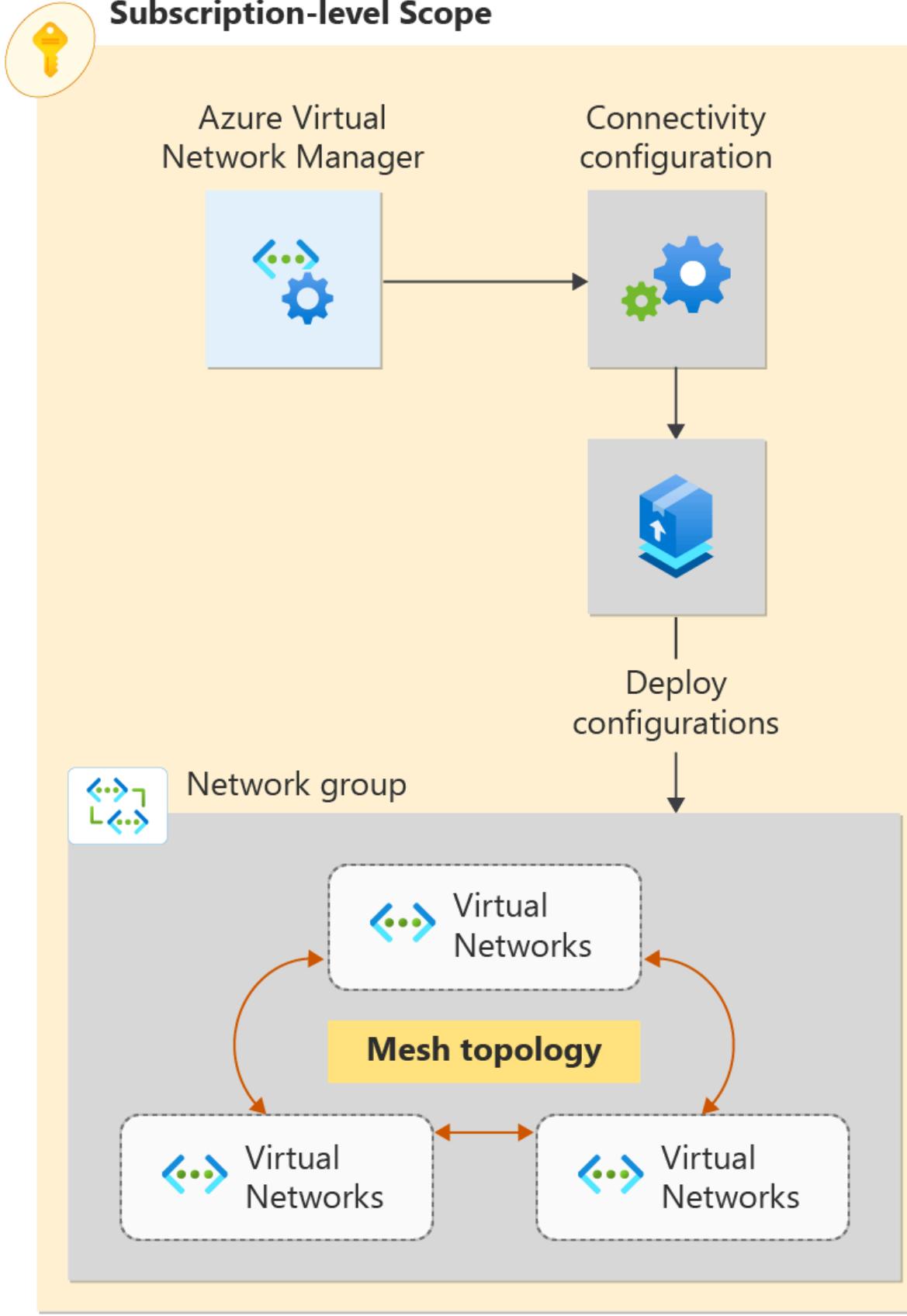
- [Azure Public DNS](#) is a hosting service for DNS domains. By hosting your domains in Azure, you can manage your DNS records by using the same credentials, APIs, tools, and billing as your other Azure services.
- [Azure Private DNS](#) is a DNS service for your virtual networks. Azure Private DNS manages and resolves domain names in the virtual network without the need to configure a custom DNS solution.
- [Azure DNS Private Resolver](#) is a service that enables you to query Azure DNS private zones from an on-premises environment and vice versa without deploying VM based DNS servers.

Using Azure DNS, you can host and resolve public domains, manage DNS resolution in your virtual networks, and enable name resolution between Azure and your on-premises resources.

Azure Virtual Network Manager

[Azure Virtual Network Manager](#) is a management service that enables you to group, configure, deploy, and manage virtual networks globally across subscriptions. With Virtual Network Manager, you can define [network groups](#) to identify and logically segment your virtual networks. Then you can determine the [connectivity](#) and [security configurations](#) you want and apply them across all the selected virtual networks in network groups at once.

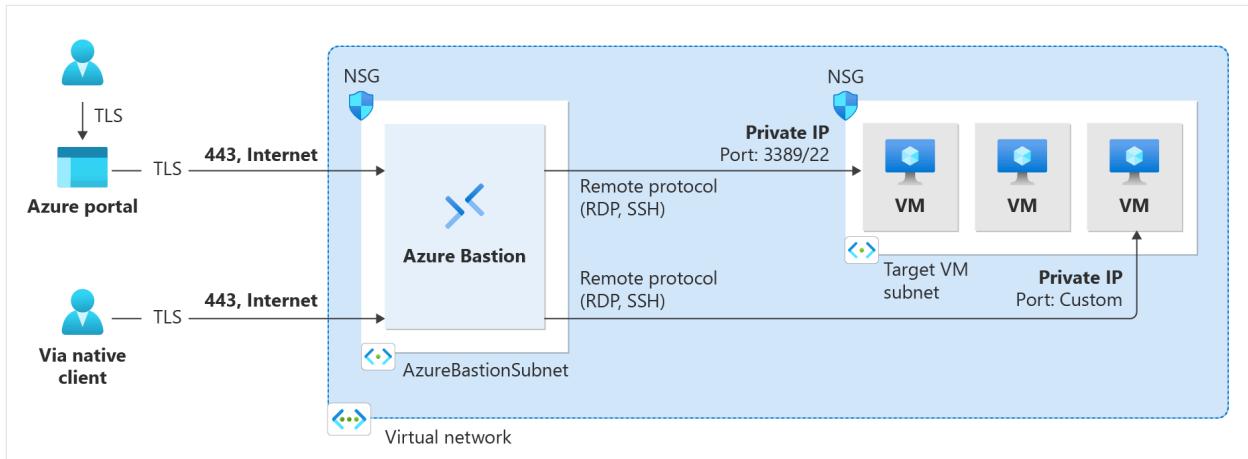
Subscription-level Scope



Azure Bastion

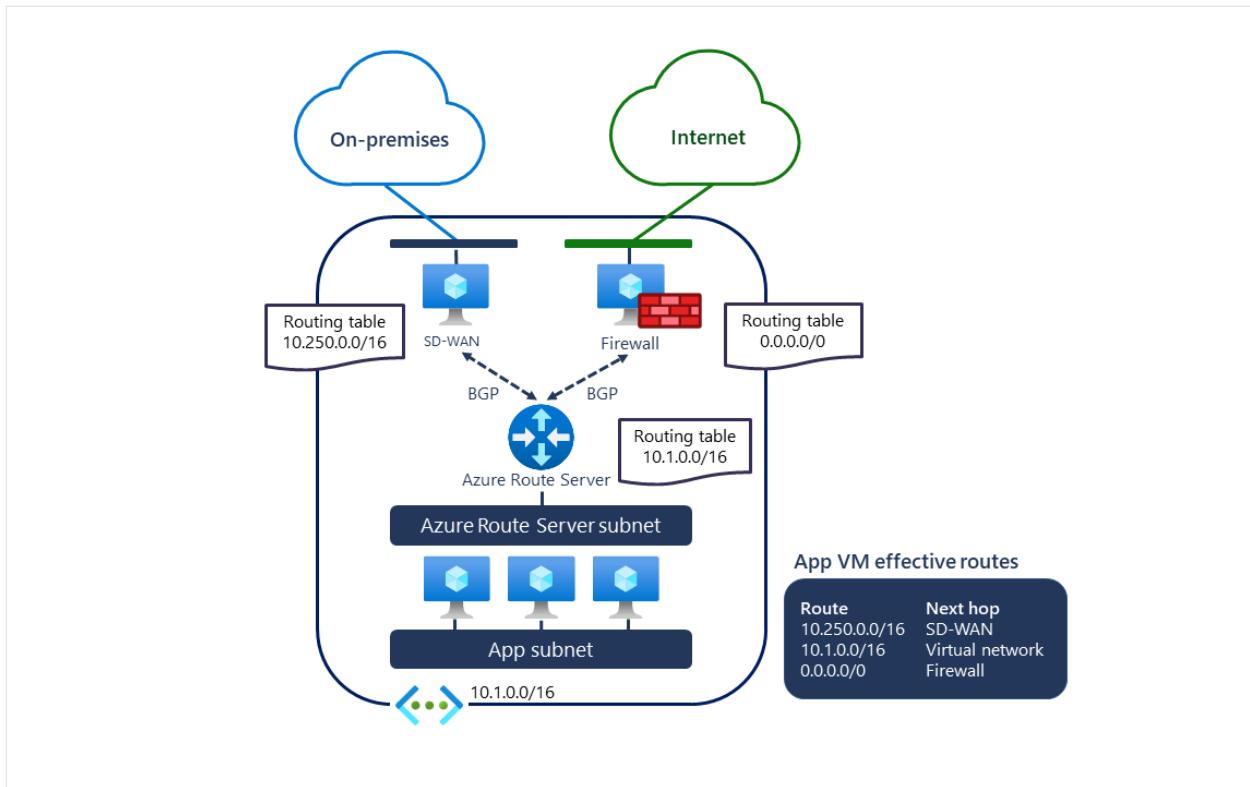
[Azure Bastion](#) is a service that you can deploy in a virtual network to allow you to connect to a virtual machine using your browser and the Azure portal. You can also connect using the native SSH or RDP client already installed on your local computer. The

Azure Bastion service is a fully platform-managed PaaS service that you deploy inside your virtual network. It provides secure and seamless RDP/SSH connectivity to your virtual machines directly from the Azure portal over TLS. When you connect via Azure Bastion, your virtual machines don't need a public IP address, agent, or special client software. There are various different SKU/tiers available for Azure Bastion. The tier you select affects the features that are available. For more information, see [About Bastion configuration settings](#).



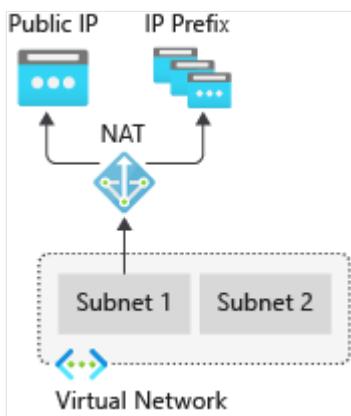
Azure Route Server

Azure Route Server simplifies dynamic routing between your network virtual appliance (NVA) and your virtual network. It allows you to exchange routing information directly through Border Gateway Protocol (BGP) routing protocol between any NVA that supports the BGP routing protocol and the Azure Software Defined Network (SDN) in the Azure Virtual Network (VNet) without the need to manually configure or maintain route tables.



NAT Gateway

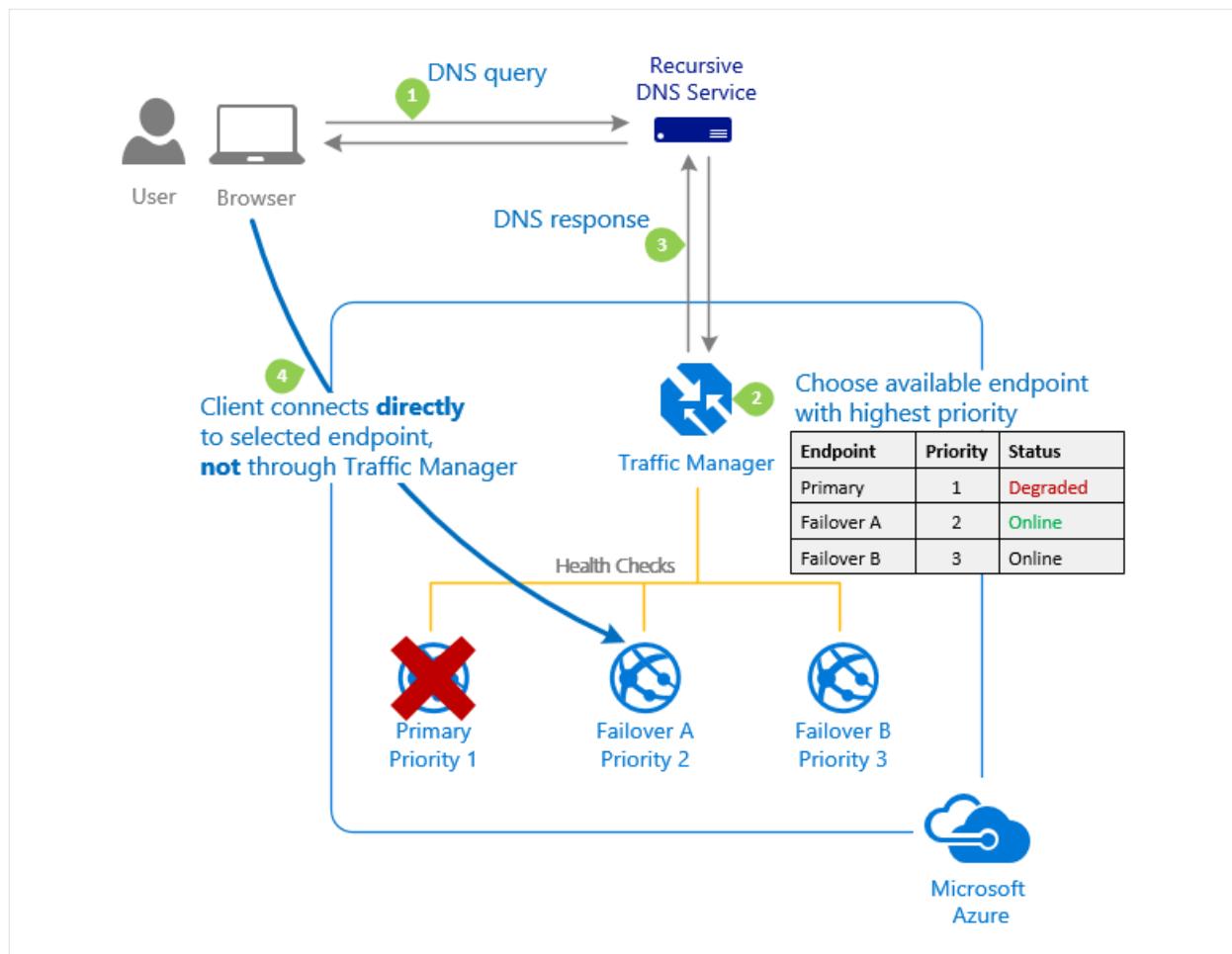
Virtual Network NAT(network address translation) simplifies outbound-only Internet connectivity for virtual networks. When configured on a subnet, all outbound connectivity uses your specified static public IP addresses. Outbound connectivity is possible without load balancer or public IP addresses directly attached to virtual machines. For more information, see [What is Azure NAT gateway?](#)



Traffic Manager

[Azure Traffic Manager](#) is a DNS-based traffic load balancer that enables you to distribute traffic optimally to services across global Azure regions, while providing high availability and responsiveness. Traffic Manager provides a range of traffic-routing methods to distribute traffic such as priority, weighted, performance, geographic, multi-value, or subnet.

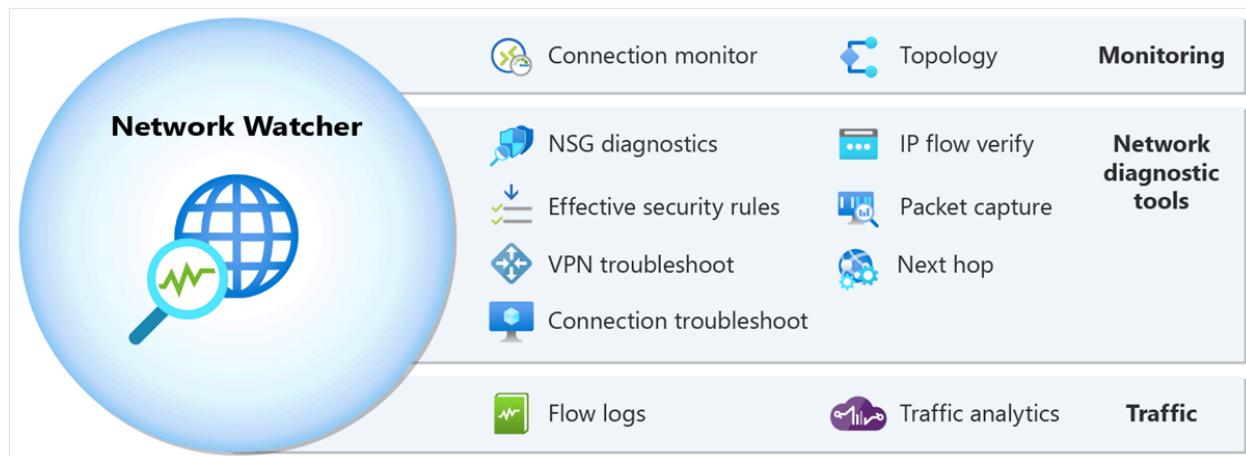
The following diagram shows endpoint priority-based routing with Traffic Manager:



For more information about Traffic Manager, see [What is Azure Traffic Manager?](#).

Azure Network Watcher

Azure Network Watcher provides tools to monitor, diagnose, view metrics, and enable or disable logs for resources in an Azure virtual network.



Azure Monitor

Azure Monitor maximizes the availability and performance of your applications by delivering a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources they depend on.

Load balancing and content delivery

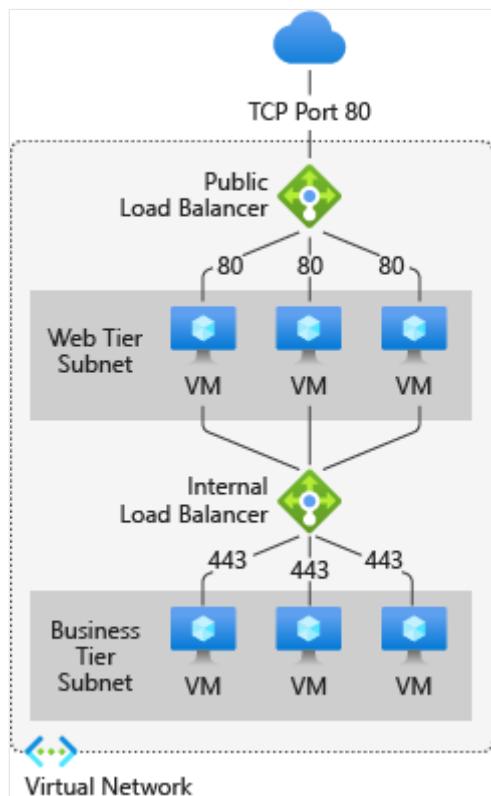
This section describes networking services in Azure that help deliver applications and workloads - Load Balancer, Application Gateway, and Azure Front Door Service.

Load Balancer

[Azure Load Balancer](#) provides high-performance, low-latency Layer 4 load-balancing for all UDP and TCP protocols. It manages inbound and outbound connections. You can configure public and internal load-balanced endpoints. You can define rules to map inbound connections to back-end pool destinations by using TCP and HTTP health-probing options to manage service availability.

Azure Load Balancer is available in Standard, Regional, and Gateway SKUs.

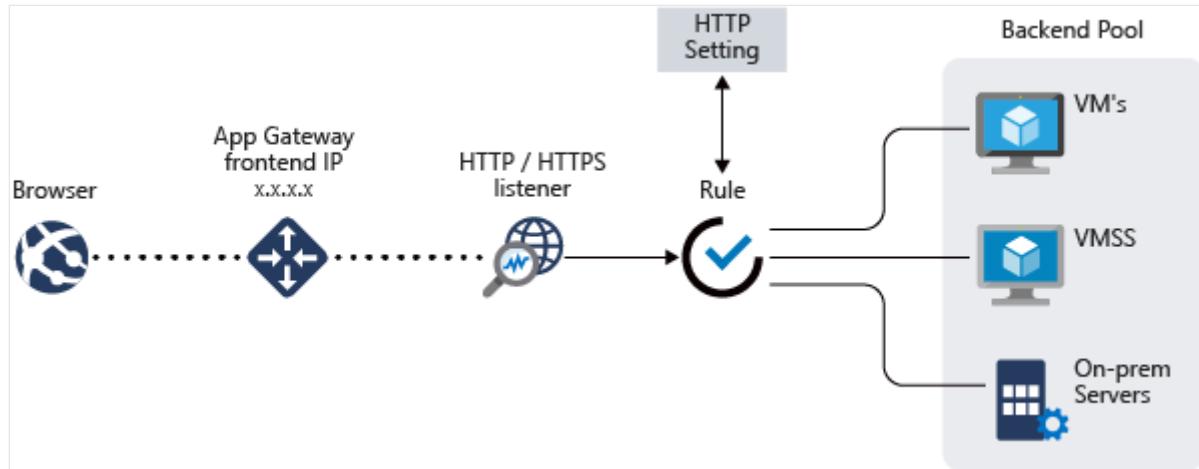
The following picture shows an Internet-facing multi-tier application that utilizes both external and internal load balancers:



Application Gateway

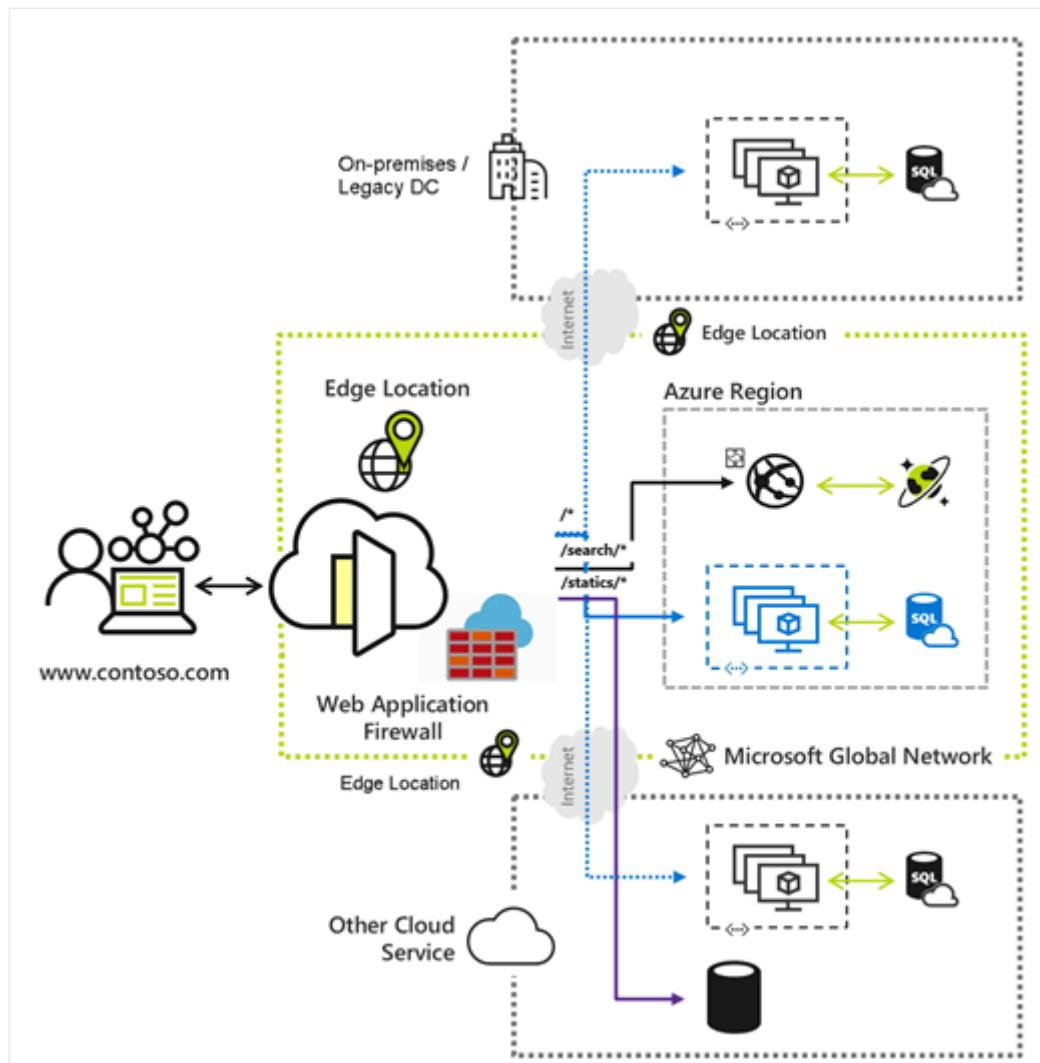
Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications. It's an Application Delivery Controller (ADC) as a service, offering various layer 7 load-balancing capabilities for your applications.

The following diagram shows url path-based routing with Application Gateway.



Azure Front Door

Azure Front Door enables you to define, manage, and monitor the global routing for your web traffic by optimizing for best performance and instant global failover for high availability. With Front Door, you can transform your global (multi-region) consumer and enterprise applications into robust, high-performance personalized modern applications, APIs, and content that reach a global audience with Azure.



Hybrid connectivity

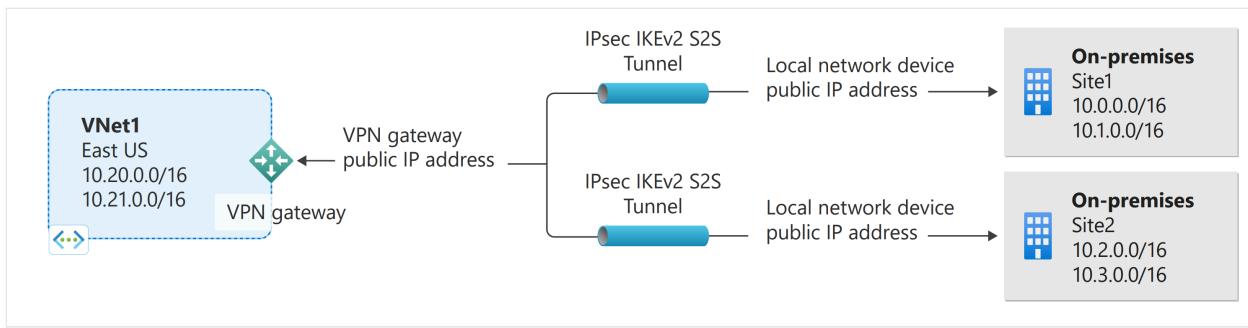
This section describes network connectivity services that provide a secure communication between your on-premises network and Azure - VPN Gateway, ExpressRoute, Virtual WAN, and Peering Service.

VPN Gateway

[VPN Gateway](#) helps you create encrypted cross-premises connections to your virtual network from on-premises locations, or create encrypted connections between VNets. There are different configurations available for VPN Gateway connections. Some of the main features include:

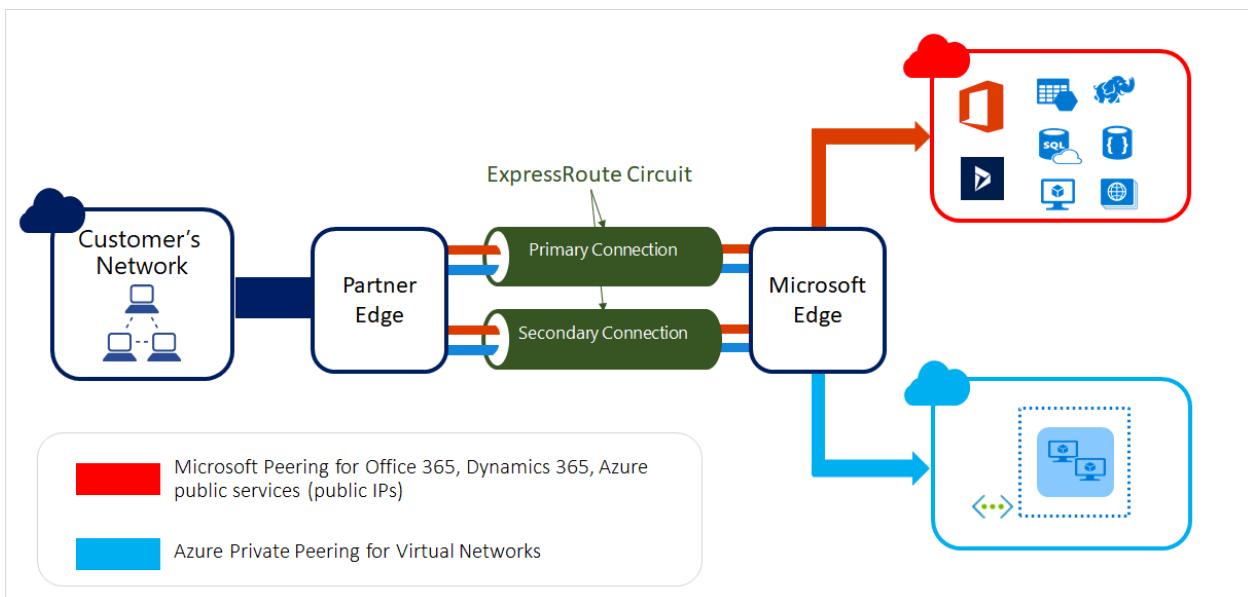
- Site-to-site VPN connectivity
- Point-to-site VPN connectivity
- VNet-to-VNet VPN connectivity

The following diagram illustrates multiple site-to-site VPN connections to the same virtual network. To view more connection diagrams, see [VPN Gateway - design](#).



ExpressRoute

[ExpressRoute](#) enables you to extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. This connection is private. Traffic doesn't go over the internet. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure, Microsoft 365, and Dynamics 365.

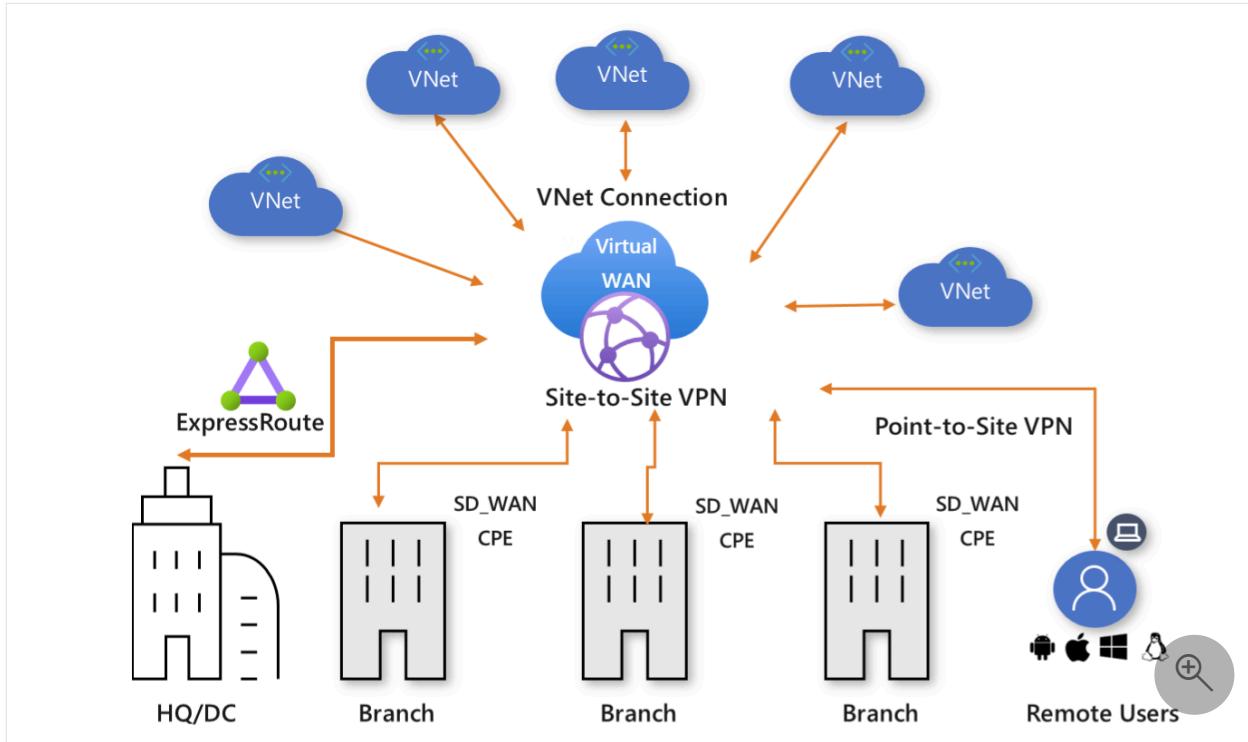


Virtual WAN

[Azure Virtual WAN](#) is a networking service that brings many networking, security, and routing functionalities together to provide a single operational interface. Connectivity to Azure VNets is established by using virtual network connections. Some of the main features include:

- Branch connectivity (via connectivity automation from Virtual WAN Partner devices such as SD-WAN or VPN CPE)
- Site-to-site VPN connectivity
- Remote user VPN connectivity (point-to-site)
- Private connectivity (ExpressRoute)
- Intra-cloud connectivity (transitive connectivity for virtual networks)

- VPN ExpressRoute inter-connectivity
- Routing, Azure Firewall, and encryption for private connectivity



Peering Service

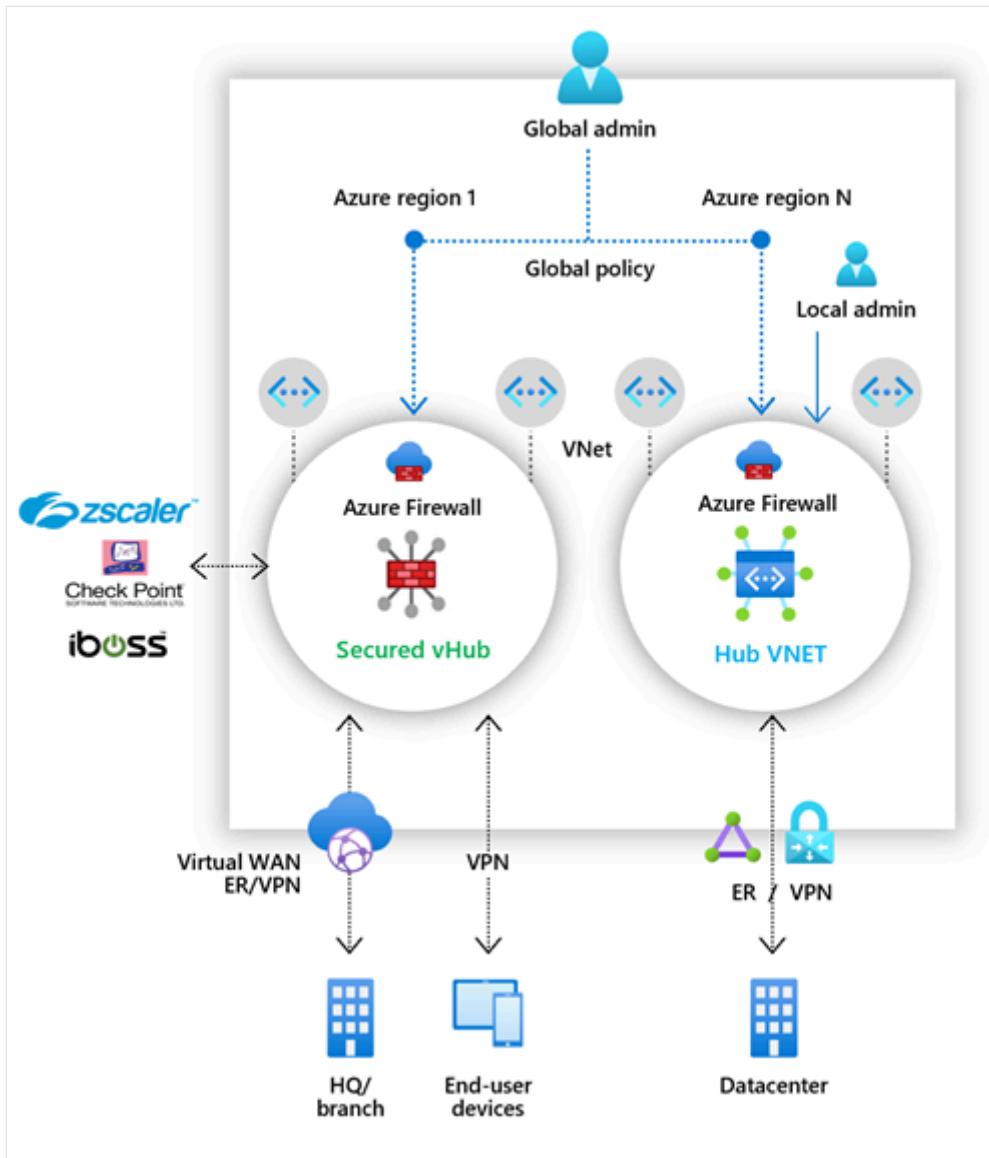
Azure Peering Service enhances customer connectivity to Microsoft cloud services such as Microsoft 365, Dynamics 365, software as a service (SaaS) services, Azure, or any Microsoft services accessible via the public internet.

Network security

This section describes networking services in Azure that protects and monitor your network resources - Firewall Manager, Firewall, Web Application Firewall, and DDoS Protection.

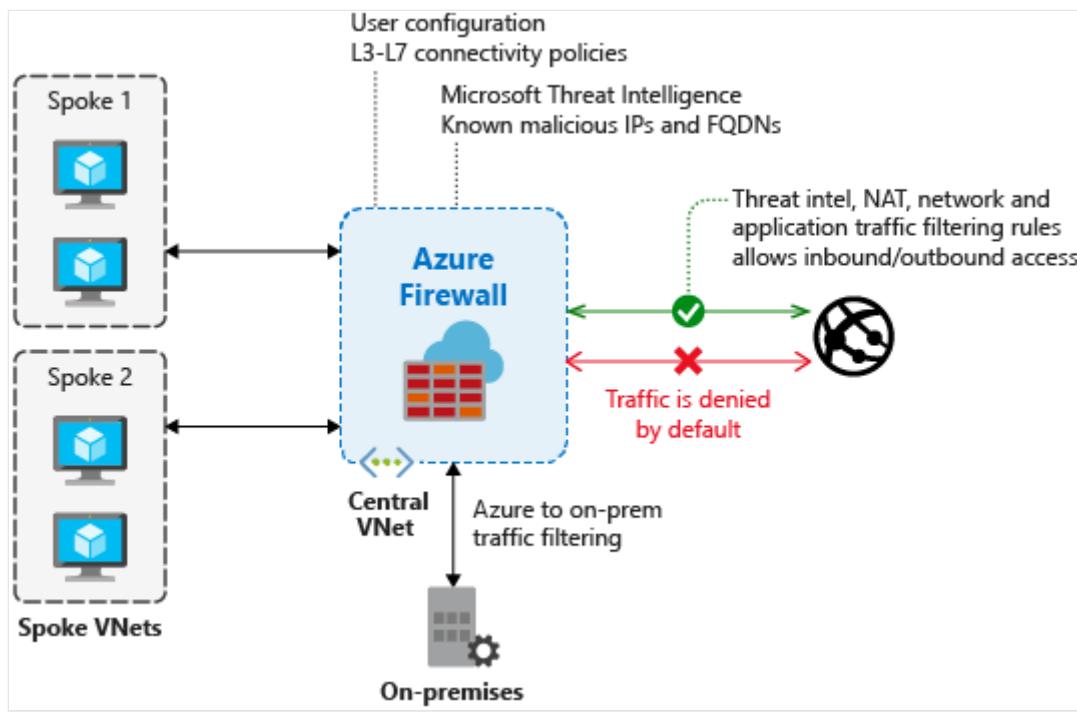
Firewall Manager

Azure Firewall Manager is a security management service that provides central security policy and routing management for cloud based security perimeters. Firewall manager can provide security management for two different types of network architecture: secure virtual hub and hub virtual network. With Azure Firewall Manager, you can deploy multiple Azure Firewall instances across Azure regions and subscriptions, implement DDoS protection plans, manage web application firewall policies, and integrate with partner security-as-a-service for enhanced security.



Azure Firewall

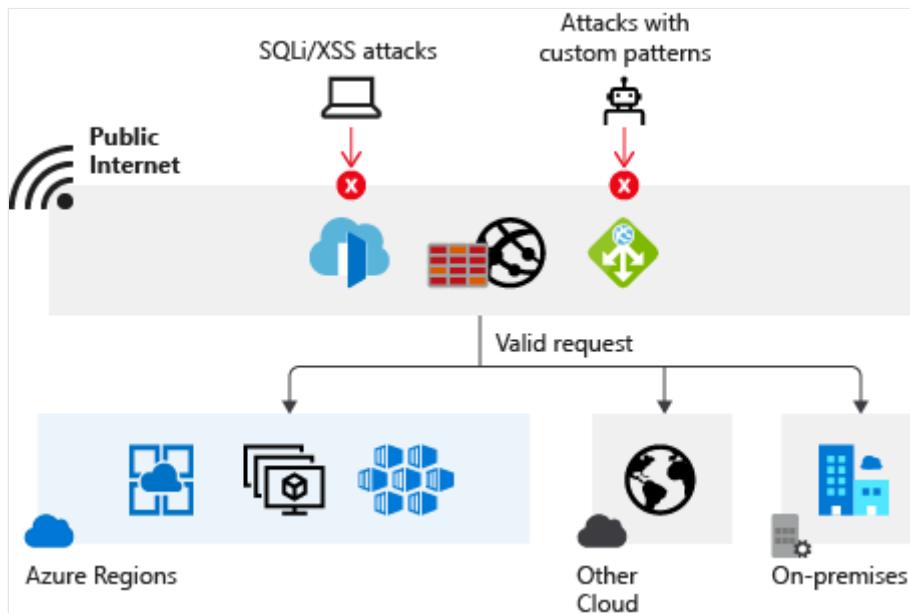
[Azure Firewall](#) is a managed, cloud-based network security service that protects your Azure Virtual Network resources. Using Azure Firewall, you can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static public IP address for your virtual network resources allowing outside firewalls to identify traffic originating from your virtual network.



Web Application Firewall

Azure Web Application Firewall (WAF) provides protection to your web applications from common web exploits and vulnerabilities such as SQL injection, and cross site scripting. Azure WAF provides out of box protection from OWASP top 10 vulnerabilities via managed rules. Additionally customers can also configure custom rules, which are customer managed rules to provide extra protection based on source IP range, and request attributes such as headers, cookies, form data fields, or query string parameters.

Customers can choose to deploy [Azure WAF with Application Gateway](#), which provides regional protection to entities in public and private address space. Customers can also choose to deploy [Azure WAF with Front Door](#) which provides protection at the network edge to public endpoints.

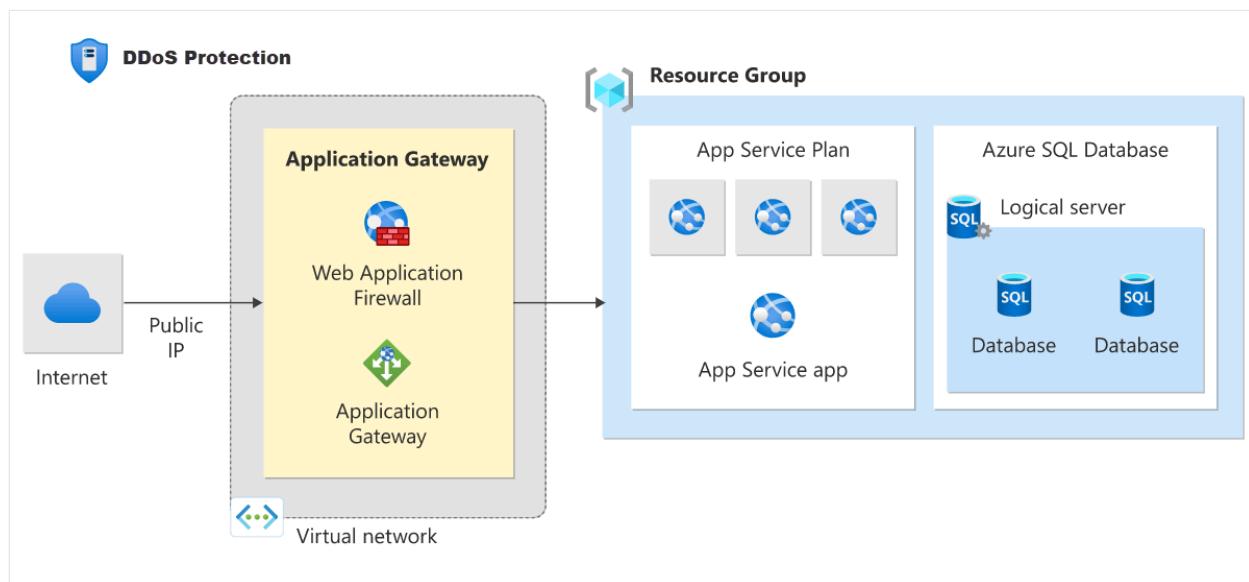


DDoS Protection

Azure DDoS Protection provides countermeasures against the most sophisticated DDoS threats. The service provides enhanced DDoS mitigation capabilities for your application and resources deployed in your virtual networks. Additionally, customers using Azure DDoS Protection have access to DDoS Rapid Response support to engage DDoS experts during an active attack.

Azure DDoS Protection consists of two tiers:

- [DDoS Network Protection](#), combined with application design best practices, provides enhanced DDoS mitigation features to defend against DDoS attacks. It's automatically tuned to help protect your specific Azure resources in a virtual network.
- [DDoS IP Protection](#) is a pay-per-protected IP model. DDoS IP Protection contains the same core engineering features as DDoS Network Protection, but differs in the following value-added services: DDoS rapid response support, cost protection, and discounts on WAF.



Next steps

- Create your first virtual network, and connect a few virtual machines to it, by completing the steps in the [Create your first virtual network](#) article.
- Connect your computer to a virtual network by completing the steps in the [Configure a point-to-site connection](#) article.
- Load balance Internet traffic to public servers by completing the steps in the [Create an Internet-facing load balancer](#) article.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

CLI samples for Azure App Service

Article • 10/12/2022

The following table includes links to bash scripts built using the Azure CLI.

[+] Expand table

Script	Description
Create app	
Create an app and deploy files with FTP	Creates an App Service app and deploys a file to it using FTP.
Create an app and deploy code from GitHub	Creates an App Service app and deploys code from a public GitHub repository.
Create an app with continuous deployment from GitHub	Creates an App Service app with continuous publishing from a GitHub repository you own.
Create an app and deploy code into a local Git repository	Creates an App Service app and configures code push into a local Git repository.
Create an app and deploy code to a staging environment	Creates an App Service app with a deployment slot for staging code changes.
Create an ASP.NET Core app in a Docker container	Creates an App Service app on Linux and loads a Docker image from Docker Hub.
Create an app with a Private Endpoint	Creates an App Service app and a Private Endpoint
Configure app	
Map a custom domain to an app	Creates an App Service app and maps a custom domain name to it.
Bind a custom TLS/SSL certificate to an app	Creates an App Service app and binds the TLS/SSL certificate of a custom domain name to it.
Scale app	
Scale an app manually	Creates an App Service app and scales it across 2 instances.
Scale an app worldwide with a high-availability architecture	Creates two App Service apps in two different geographical regions and makes them available through a single endpoint

Script	Description
	using Azure Traffic Manager.
Protect app	
Integrate with Azure Application Gateway	Creates an App Service app and integrates it with Application Gateway using service endpoint and access restrictions.
Connect app to resources	
Connect an app to a SQL Database	Creates an App Service app and a database in Azure SQL Database, then adds the database connection string to the app settings.
Connect an app to a storage account	Creates an App Service app and a storage account, then adds the storage connection string to the app settings.
Connect an app to an Azure Cache for Redis	Creates an App Service app and an Azure Cache for Redis, then adds the redis connection details to the app settings.)
Connect an app to Azure Cosmos DB	Creates an App Service app and an Azure Cosmos DB, then adds the Azure Cosmos DB connection details to the app settings.
Backup and restore app	
Backup and restore app	Creates an App Service app and creates a one-time backup for it, creates a backup schedule for it, and then restores an App Service app from a backup.
Monitor app	
Monitor an app with web server logs	Creates an App Service app, enables logging for it, and downloads the logs to your local machine.

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback ↗](#) | [Get help at Microsoft Q&A](#)

Azure command line tools survey guidance

Article • 06/26/2023

When using the Azure CLI, you may be invited to participate in a survey to tell us about your experience. By responding to the survey, you help to identify common issues and areas for improvement. Understanding your experiences and opinions helps to make future releases of Azure command line tools better for you and others.

What data is collected

The survey collects anonymized feedback about your satisfaction with Azure command line tools and doesn't collect any private or personal data.

While we appreciate the insights this data provides, we understand not everyone wants to be prompted to complete a survey. You can disable being prompted to participate in surveys with the `az config` command or via an environment variable.

Disable the survey

In the following example, the `az config` command is used to disable the survey message.

Azure CLI

```
az config set core.survey_message=false
```

You can also use the `AZURE_CORE_SURVEY_MESSAGE` environment variable to disable the survey message.

Privacy statement

Your privacy is important to us. [Microsoft's Privacy Statement](#) explains the personal data Microsoft processes, how Microsoft processes it, and for what purposes.

Troubleshooting Azure CLI

Article • 07/24/2024

Error categories

Most errors returned by the Azure CLI fall into one of these categories:

[+] Expand table

Error category	General error cause
Unrecognized argument	A parameter is misspelled or doesn't exist.
Required argument missing	A required parameter isn't specified or only one of two "parameter pairs" is specified. A parameter might also be misspelled.
Mutually exclusive argument	Two or more parameters can't be specified together.
Invalid argument value	Parameter <i>value</i> isn't valid. This error is often due to quoting, an escape character or spacing.
Bad request	An HTTP status code of 400 returns this error. Check for a missing space, missing parameter dash, or an extra or missing single or double quotation mark. This error also happens when a parameter value doesn't contain an allowed value.
Resource not found	An Azure resource referenced in a parameter value can't be found.
Authentication	Microsoft Entra authentication failed.

The `--debug` parameter

One of the best ways to see what the Azure CLI is executing for each Azure CLI reference command is to use the `--debug` parameter. Here's examples of `--debug` for both a failed and successful command:

```
Azure CLI

# Error example: Create a resource group, but omit the quotes around the
resource group name.
az group create --location eastus2 --name msdocs-rg-test --debug
```

Here's a portion of the debug output. Notice the log location and unrecognized argument.

Output

```
cli.knack.cli: Command arguments: ['group', 'create', '-l', 'eastus2', '-name', 'msdocs-rg-test', '--debug']
...
cli.azure.cli.core.azlogging: metadata file logging enabled - writing logs
to '/home/myName/.azure/commands/YYYY-MM-DD.HH-MM-SS.group_create.8912.log'.
...
cli.azure.cli.core.azclierror: unrecognized arguments: msdocs-rg-test
...
```

Compare the error `--debug` output given above to a successful execution:

Azure CLI

```
# Correct example: Because the resource group name contains special
# characters, enclose it in quotes
az group create --location eastus2 --name "msdocs-rg-test" --debug
```

Here's a portion of the debug output. Notice the log location, API call, and run time.

Output

```
cli.knack.cli: Command arguments: ['group', 'create', '-l', 'eastus2', '-n',
'mdocs-rg-test', '--debug']
...
cli.azure.cli.core.azlogging: metadata file logging enabled - writing logs
to '/home/myName/.azure/commands/YYYY-MM-DD.HH-MM-SS.group_create.8912.log'.
...
cli.azure.cli.core.sdk.policies: Request URL:
'https://management.azure.com/subscriptions/00000000-0000-0000-0000-
000000000000/resourcegroups/msdocs-rg-test?api-version=YYYY-MM-DD'
cli.azure.cli.core.sdk.policies: Request method: 'PUT'
cli.azure.cli.core.sdk.policies: Request headers:
cli.azure.cli.core.sdk.policies:     'Content-Type': 'application/json'
cli.azure.cli.core.sdk.policies:     'Content-Length': '23'
cli.azure.cli.core.sdk.policies:     'Accept': 'application/json'
cli.azure.cli.core.sdk.policies:     'x-ms-client-request-id': 'ba7ee6f4-
2dcc-11ef-81ce-00155dadcc5c8'
cli.azure.cli.core.sdk.policies:     'CommandName': 'group create'
cli.azure.cli.core.sdk.policies:     'ParameterSetName': '-l -n --debug'
cli.azure.cli.core.sdk.policies:     'User-Agent': 'AZURECLI/2.61.0 (RPM)
azsdk-python-core/1.28.0 Python/3.9.19 (Linux-5.10.102.2-microsoft-standard-
x86_64-with-glibc2.35) cloud-shell/1.0'
cli.azure.cli.core.sdk.policies:     'Authorization': '*****'
cli.azure.cli.core.sdk.policies: Request body:
cli.azure.cli.core.sdk.policies: {"location": "eastus2"}
```

```
urllib3.connectionpool: Starting new HTTPS connection (1):
management.azure.com:443
urllib3.connectionpool: https://management.azure.com:443 "PUT
/subscriptions/3618afcd-ea52-4ceb-bb46-53bb962d4e0b/resourcegroups/msdocs-
rg-test?api-version=2022-09-01 HTTP/1.1" 201 226
cli.azure.cli.core.sdk.policies: Response status: 201
...
cli.azure.cli.core.sdk.policies:      'Date': 'Tue, 18 Jun 2024 23:44:41 GMT'
cli.azure.cli.core.sdk.policies: Response content:
cli.azure.cli.core.sdk.policies: {"id":"/subscriptions/00000000-0000-0000-
0000-000000000000/resourceGroups/msdocs-rg-test","name":"msdocs-rg-
test","type":"Microsoft.Resources/resourceGroups","location":"eastus2","prop
erties":{"provisioningState":"Succeeded"}}
...
cli.__main__: Command ran in 1.829 seconds (init: 0.111, invoke: 1.718)
```

For examples of `--debug` for JSON formatting, see [Quoting differences between scripting languages - JSON strings](#).

Common syntax errors

Although the Azure CLI can run in both Bash, PowerShell and Windows Cmd, there are syntax differences between scripting languages. Azure CLI scripts containing single quotes, double quotes, and escape characters usually must be modified when copied between languages. This challenge reveals itself most often in parameter values, especially in values assigned to the `--query` parameter. Here are some common error messages:

- "**Bad request ...{something} is invalid**" might be caused by a space, single or double quotation mark, or lack of a quote.
- "**Unexpected token...**" is seen when there's an extra space or quote.
- "**Invalid jmespath_type value**" error often comes from incorrect quoting in the `--query` parameter.
- "**Variable reference is not valid**" is received when a string isn't formatted properly often due to concatenation or a missing escape character.
- "**Unrecognized arguments**" is often caused by an incorrect line continuation character or misspelled parameter name.
- "**Missing expression after unary operator**" is seen when a line continuation character is missing.

There are several Azure CLI articles dedicated to explaining syntax errors and providing working examples:

- [Quoting differences between scripting languages](#)
- [Syntax differences in Bash, PowerShell, and Cmd tutorial](#)
- Find many `--query` parameter examples in [How-to query Azure CLI command output using a JMESPath query](#)
- [How-to use the Azure CLI in a Bash scripting language](#)
- [Considerations for running the Azure CLI in a PowerShell scripting language](#)

💡 Tip

If you can't resolve a command error, try using a different scripting language. **Most Azure CLI documentation is written and tested in Azure Cloud Shell (ACS) with a Bash scripting language.** If you can get an article example to execute in ACS Bash, but it won't execute in Windows PowerShell, review your use of single and double quotes, and escape characters.

Error: Invalid value or doesn't exist

These errors often occur when trying to use a variable value that contains an incorrect format. The default output for Azure CLI is JSON, so if you're trying to store an ID for an Azure resource in a variable, you must specify `--output tsv`. Here's an example:

```
Azure CLI

# Get a subscription that contains a name or phrase
subscriptionID=$(az account list --query "[?contains(name,'my case sensitive
search phrase')].id")
echo $subscriptionID

# output as JSON
[ "00000000-0000-0000-0000-000000000000" ]

# Try to set your subscription to the new ID
az account set --subscription $subscriptionID

# error output
The subscription of '"00000000-0000-0000-0000-000000000000"' doesn't exist
in cloud 'AzureCloud'.
```

Now use the `tsv` output type.

```
Azure CLI
```

```
# Get the active subscription
subscriptionID=$(az account list --query "[?contains(name,'my case sensitive
search phrase')].id" --output tsv)
echo $subscriptionID

# output as TSV
00000000-0000-0000-0000-000000000000

# Successfully set your subscription to the new ID
az account set --subscription $subscriptionID
```

Error: Arguments are expected or required

You receive this error when an Azure CLI command is missing a required parameter, or *there's a typographical error that causes the Azure CLI to incorrectly parse the reference command*. When working with a script, you also receive this error when one or more conditions are true:

- A line continuation character is missing or incorrect.
- A trailing space exists on the right side of a line continuation character when working in the PowerShell scripting language. At this time, [splatting](#) isn't supported with Azure CLI commands.
- A variable name contains a special character, such as a dash (-).

Error: Resource not found

When the Azure CLI can't find the resource name or ID passed in a parameter value, it's usually because of one of these reasons:

- The resource name or ID is spelled incorrectly.
- The resource name contains special characters and isn't surrounded by single or double quotes.
- The value being passed to a variable has unseen leading or trailing spaces.
- The resource exists, but is in a different subscription.

Error: Failed to parse string as JSON

There are quoting differences between Bash, PowerShell in Linux, and PowerShell in Windows. Furthermore, different versions of PowerShell can produce different results. For complex parameters, like a JSON string, the best practice is to use Azure CLI's

@<file> convention to bypass a shell's interpretation. For more information, see one of these articles:

For JSON syntax examples for Bash, PowerShell and Cmd.exe, see [Quoting differences between scripting languages - JSON strings](#) tutorial.

Error: InvalidTemplateDeployment

When you try to create an Azure resource in a location that doesn't offer that resource, you receive an error similar to this message: "Following SKUs have failed for Capacity Restrictions: myDesiredSkuName' is currently not available in location 'mySpecifiedLocation'."

Here's a full error example for a VM that can't be created in the `westus` location:

```
Azure CLI

{"error": {"code": "InvalidTemplateDeployment", "message": "The template deployment 'vm_deploy_<32 character ID>' is not valid according to the validation procedure. The tracking id is '<36 character ID>'. See inner errors for details.", "details": [{"code": "SKUNotAvailable", "message": "The requested VM size for resource 'Following SKUs have failed for Capacity Restrictions: Standard_DS1_v2' is currently not available in location 'westus'. Please try another size or deploy to a different location or different zone. See https://aka.ms/azureskunotavailable for details."}]}}
```

The solution is to change a property of your requested Azure resource, or try a different location.

Error: Subscription not found

Assuming that you have not incorrectly typed a subscription name or ID, this error occurs when a resource provider is not registered in the active subscription. For example, if you want to execute `az storage account create`, the `Microsoft.Storage` provider must be registered. To register a resource provider, see [Azure resource providers and types](#).

Error: Bad handshake...certificate verify failed

See [Work behind a proxy](#) for information on how to resolve this error.

Work behind a proxy

If you're using Azure CLI over a proxy server that uses self-signed certificates, the Python [requests library](#) used by the Azure CLI might cause the following error: `SSLError("bad handshake: Error([('SSL routines', 'tls_process_server_certificate', 'certificate verify failed')],))`. To address this error, set the environment variable `REQUESTS_CA_BUNDLE` to the path of CA bundle certificate file in PEM format.

[+] [Expand table](#)

OS	Default certificate authority bundle
Windows 32-bit	<code>C:\Program Files (x86)\Microsoft SDKs\Azure\CLI2\Lib\site-packages\certifi\cacert.pem</code>
Windows 64-bit	<code>C:\Program Files\Microsoft SDKs\Azure\CLI2\Lib\site-packages\certifi\cacert.pem</code>
Ubuntu/Debian Linux	<code>/opt/az/lib/python<version>/site-packages/certifi/cacert.pem</code>
CentOS Stream/RHEL/SUSE Linux	<code>/usr/lib64/az/lib/python<version>/site-packages/certifi/cacert.pem</code>
macOS	<code>/usr/local/Cellar/azure-cli/<cliversion>/libexec/lib/python<version>/site-packages/certifi/cacert.pem</code>

Append the proxy server's certificate to the CA bundle certificate file, or copy the contents to another certificate file. Then set `REQUESTS_CA_BUNDLE` to the new file location. Here's an example:

```
Console

<Original cacert.pem>

-----BEGIN CERTIFICATE-----
<Your proxy's certificate here>
-----END CERTIFICATE-----
```

Some proxies require authentication. The format of the `HTTP_PROXY` or `HTTPS_PROXY` environment variables should include the authentication, such as

`HTTPS_PROXY="https://username:password@proxy-server:port"`. For details, see [How to configure proxies for the Azure SDK for Python](#).

Service principals

For information on troubleshooting service principals, see [Cleanup and Troubleshooting](#) in the [Work with service principals](#) tutorial.

Other issues

If you experience a product issue with Azure CLI not listed in this article, [file an issue on GitHub](#).

See also

- [Tips for using the Azure CLI successfully](#)

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



Azure CLI feedback

Azure CLI is an open source project. Select a link to provide feedback:

-  [Open a documentation issue](#)
-  [Provide product feedback](#)